National Research Council Canada
Conseil national de recherches Canada

NRC

A KALMAN FILTER INTEGRATED NAVIGATION DESIGN FOR THE IAR TWIN OTTER ATMOSPHERIC RESEARCH AIRCRAFT

by

B.W. Leach

Institute for Aerospace Research

OTTAWA
APRIL 1991

AERONAUTICAL NOTE
IAR-AN-72
NRC NO. 32148

Canada

UNLIMITED
UNCLASSIFIED

INSTITUTE FOR AEROSPACE RESEARCH

SCIENTIFIC AND TECHNICAL PUBLICATIONS

AERONAUTICAL REPORTS

Aeronautical Reports (LR): Scientific and technical information pertaining to aeronautics considered important, complete, and a lasting contribution to existing knowledge.

Mechanical Engineering Reports (MS): Scientific and technical information pertaining to investigations outside aeronautics considered important, complete, and a lasting contribution to existing knowledge.

AERONAUTICAL NOTES (AN): Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

LABORATORY TECHNICAL REPORTS (LTR): Information receiving limited distribution because of preliminary data, security classification, proprietary, or other reasons.

Details on the availability of these publications may be obtained from:

Graphics Section,
National Research Council Canada,
Institute for Aerospace Research,
Bldg. M-16, Room 204,
Montreal Road,
Ottawa, Ontario
K1A 0R6

INSTITUT DE RECHERCHE AÉROSPATIALE

PUBLICATIONS SCIENTIFIQUES ET TECHNIQUES

RAPPORTS D'AÉRONAUTIQUE

Rapports d'aéronautique (LR): Informations scientifiques et techniques touchant l'aéronautique jugées importantes, complètes et durables en termes de contribution aux connaissances actuelles.

Rapports de génie mécanique (MS): Informations scientifiques et techniques sur la recherche externe à l'aéronautique jugées importantes, complètes et durables en termes de contribution aux connaissances actuelles.

CAHIERS D'AÉRONAUTIQUE (AN): Informations de moindre portée mais importantes en termes d'accroissement des connaissances.

RAPPORTS TECHNIQUES DE LABORATOIRE (LTR): Informations peu disséminées pour des raisons d'usage secret, de droit de propriété ou autres ou parce qu'elles constituent des données préliminaires.

Les publications ci-dessus peuvent être obtenues à l'adresse suivante:

Section des graphiques,
Conseil national de recherches Canada,
Institut de recherche aérospatiale,
Im. M-16, pièce 204,
Chemin de Montréal,
Ottawa (Ontario)
K1A 0R6

UNLIMITED
UNCLASSIFIED

A KALMAN FILTER INTEGRATED NAVIGATION DESIGN FOR THE IAR TWIN OTTER ATMOSPHERIC RESEARCH AIRCRAFT

MÉTHODE DE NAVIGATION INTÉGRÉE À FILTRE DE KALMAN DESTINÉE AU TWIN OTTER DE L'IRA CHARGÉ DES RECHERCHES ATMOSPHÉRIQUES

by

B.W. Leach

Institute for Aerospace Research

OTTAWA
APRIL 1991

AERONAUTICAL NOTE
IAR-AN-72
NRC NO. 32148

S.R.M. Sinclair, Head/Chef
Flight Research Laboratory/
Laboratoire de recherches en vol

G.F. Marsters
Director General
Le directeur général

91-04736/

SUMMARY

The IAR Twin Otter Atmospheric Research Aircraft has a continuing requirement for more accurate, inertially-based navigation data for both track recovery and the calculation of wind gust components. This navigational accuracy is necessary, not just during post-flight analysis, but also for real-time, in-flight guidance and wind computation. Previous developmental work at the Flight Research Laboratory on advanced navigation systems has demonstrated the benefits of a Kalman filter integrated navigation approach in order to satisfy the most stringent navigational requirements. A significant upgrade to the navigation sensor suite onboard the Twin Otter in the last two years has resulted in the potential, via Kalman filtering, for generating very high quality inertial velocity and positional information in real time, together with improved airborne wind components.

The Kalman filter integrated navigation design described in this report is based on the optimal blending of data from an LTN-90-100 strapdown Inertial Reference System (IRS), a Decca Type 72 Doppler velocity sensing (DVS) system and an ARNAV R-40 airborne Loran-C receiver - sensors that are available on the Twin Otter at the present time. In the Twin Otter's real-time computing/data acquisition system, all three of these navigation sensors are interfaced to the onboard LSI-11/73 microcomputer system, and a complete set of navigation parameters is being recorded. In particular, all of the raw inertial data parameters from the LTN-90-100 IRS, required for proper design of an IRS-based Kalman filter, are available with sufficient resolution and at a suitable digital sampling rate.

A driving force behind the decision to pursue this integrated navigation approach on the Twin Otter has been the observation that significant velocity errors (and, eventually, position errors) can occur in the LTN-90-100 IRS over the course of a flight, and the observed error levels can seriously degrade the accuracy of the wind calculations. On the other hand, the airborne Loran-C positional data has been demonstrated to be consistently more accurate than the IRS position information, in the long term. An integrated navigation system approach, using the principles of Kalman filtering, is shown to have the ability to use Loran-C data (and, to a lesser extent, Doppler velocity data) to accurately track the dominant IRS errors (position, velocity and attitude components), and provide IRS error corrections at a rate appropriate for Twin Otter requirements.

RÉSUMÉ

Le Twin Otter de l'IRA chargé des recherches atmosphériques nécessite constamment une plus grande précision des données de navigation par inertie, tant pour le rétablissement de la trajectoire que pour le calcul des composantes des rafales de vent. La précision de la navigation est nécessaire non seulement lors de l'analyse consécutive à un vol, mais également lors du guidage en vol en temps réel et du calcul du vent. Au Laboratoire de recherche en vol, les systèmes avancés de navigation ont fait l'objet de travaux antérieurs de mise au point qui ont démontré qu'un dispositif de navigation intégrée à filtre Kalman présentait des avantages permettant de satisfaire aux exigences de navigation les plus sévères. Au cours des deux dernières années, une amélioration importante de l'ensemble des détecteurs installés à bord du Twin Otter s'est traduite par la possibilité de générer, par le biais d'un filtre de Kalman, des données de très haute qualité sur la vitesse inertielle et à la position en temps réel et par l'amélioration du calcul des composantes du vent en cours de vol.

Les principes de navigation intégrée à filtre de Kalman décrits dans le présent rapport sont fondés sur la combinaison optimale de données provenant d'un système de navigation par inertie (SNI) à composants liés LTN-90-100, d'un système Doppler de mesure de vitesse Decca Type 72 et d'un récepteur Loran-C de bord ARNAV R-40; tous ces détecteurs sont présentement installés dans le Twin Otter. Le système de traitement en temps réel et d'acquisition des données du Twin Otter est formé par l'interfaçage de ces trois détecteurs de navigation au micro-ordinateur LSI 11/73 de bord; le système enregistre donc un jeu complet de paramètres de navigation. On dispose plus particulièrement de tous les paramètres d'inertie bruts du SNI LTN-90-100 nécessaires à la conception du type approprié de filtre de Kalman fondé sur SNI, la résolution et le taux d'échantillonnage numérique étant suffisants.

L'une des principales justifications de l'adoption de cette approche de navigation intégrée sur le Twin Otter est la possibilité de production pendant un vol d'importantes erreurs dans les données de vitesse (et, éventuellement, dans celles de position) du SNI LTN-90-100; les niveaux d'erreurs observés peuvent réduire substantiellement l'exactitude des calculs du vent. Par contre, on a démontré que les données de position du Loran C de bord sont généralement plus exactes à long terme que les données de position du SNI. On montre qu'une approche de navigation intégrée intégrant les principes du filtrage de Kalman peut utiliser les données Loran-C (et, dans une moindre mesure, les données de vitesse Doppler) pour suivre avec précision les principales erreurs SNI (composantes de position, vitesse, et attitude) et produire des corrections d'erreur SNI à un débit satisfaisant aux exigences du Twin Otter.

TABLE OF CONTENTS

Page

SUMMARY iii

TABLES vi

APPENDICES vi

ILLUSTRATIONS vi

SYMBOLS x

1.0 INTRODUCTION 1

2.0 THE KALMAN FILTER EQUATIONS FOR INTEGRATED NAVIGATION 2

2.1 The Linear Discrete-Time Kalman Filter. 2
2.2 Software for Performing Discrete-Time Kalman Filtering 3
2.3 Integrated Navigation Using Kalman Filtering 6

3.0 IRS/DOPPLER/LORAN-C KALMAN FILTER NAVIGATOR 7

3.1 Navigation Sensor Descriptions. 7
3.2 Overview of the Kalman Filter Navigator 7
3.3 Details of the Kalman Filter Design 9
3.3.1 Error States Chosen 9
3.3.2 Plant Dynamics (Φ, **G**, **Q**). 10
3.3.3 Measurement Process (**H**, **R**) 11

3.4 Measurement Averaging (Prefiltering) 14
3.5 Euler Angle Error Correction 15

4.0 SOFTWARE FOR NAVIGATION SIMULATION AND KALMAN FILTERING 18

4.1 Software for Generating Reference Trajectory Profile Data. 18
4.2 Software for Generating Realistic Navigation Data. 20
4.3 Kalman Filter Software Descriptions 22
4.4 Kalman Filter Results Using Simulated Data 26
4.5 Conclusions from the Simulation Studies 29

5.0 KALMAN FILTER RESULTS USING REAL FLIGHT DATA. 30

5.1 Specialized Navigation Flights 30
5.2 Fundamental Filter Performance Using Real Flight Data 31
5.2.1 Data from Flight # 1 (May 1988) 31
5.2.2 Data from Flight # 2 (May 1989) 32

5.3 A Comparison of Three Different Measurement Scenarios. 33
5.4 Conclusions from the Analysis of Real Flight Data. 33

6.0 CONCLUSIONS AND FUTURE WORK 34

7.0 ACKNOWLEDGEMENTS 35

8.0 REFERENCES 36

TABLES

Table		Page
3.1	Digital Data Parameters Available from LTN-90-100 IRS	8
3.2	Specifications for IRS Accelerometers and Gyroscopes	8
3.3	Digital Data Parameters Available from Decca Doppler System	9
3.4	IRS System Error States and Statistics	10
3.5	Markov Error States and Statistics	11
3.6	Plant Noise Components and Statistics	12
4.1	Error Specifications for LTN-90-100 IRS Simulation	27
4.2	Error Specifications for Loran-C and Doppler Simulations	28
5.1	Geographical Locations of the Visual On-tops	30

APPENDICES

Appendix		Page
A	Fundamental Inertial Navigation Definitions and Equations	38
B	Error State Equations for the IRS/Doppler/Loran-C Kalman Filter	42
C	Kalman Filter Measurement Processing	47
D	Measurement Averaging (Prefiltering)	54

ILLUSTRATIONS

Figure		Page
1	IAR Twin Otter Atmospheric Research Aircraft as Instrumented for Airborne Wind Measurement	57
2	Error State Feedforward Kalman Filter Navigator	58
3	IRS/Doppler/Loran-C Kalman Filter Configuration	59
4	Velocity Components of Simulated Trajectory	60
5	Velocity Rate Components of Simulated Trajectory	61
6	Attitude Components of Simulated Trajectory	62
7	Attitude Rate Components of Simulated Trajectory	63

Figure		Page
8	Position Change Components of Simulated Trajectory	64
9	Strapdown Navigator Position Errors	65
10	Strapdown Navigator Velocity Errors	65
11	Strapdown Navigator Attitude Errors	66
12	Loran-C Latitude and Longitude Markov Processes	66
13	Doppler Velocity Markov Processes	67
14	Listing of File **SDKFNAV CTRL**	67
15	Kalman Filter Loran-C Position Measurements	68
16	Kalman Filter Doppler Velocity Measurements	69
17	Accuracy of IRS Position Error State Estimation	70
18	Accuracy of IRS Velocity Error State Estimation	70
19	IRS Tilt and Vertical Acceleration Error State Estimates	71
20	IRS Gyro Drift Error State Estimates	71
21	IRS Accelerometer and Barometric Altimeter Error State Estimates	72
22	Accuracy of Loran-C Position and Doppler Sea Bias Error State Estimation	72
23	Accuracy of Doppler Velocity Error State Estimation	73
24	Innovations for Kalman Filter Position Measurements	73
25	Innovations for Kalman Filter Velocity Measurements	74
26	Horizontal Track Plot for Flight #2	75
27	Loran-C Position Measurements for Flight #1 – Loran-C Data Problems	76
28	Loran-C Position Measurements for Flight #1 – Problems Corrected	76
29	Doppler Velocity Measurements for Flight #1	77
30	IRS Position Errors at VOT's for Flight #1	78
31	Loran-C Position Errors at VOT's for Flight #1	78
32	**SDKFNAV CTRL** File for Flight #1	79

Figure		Page
33	IRS Position Error State Estimates for Flight #1	80
34	IRS Velocity Error State Estimates for Flight #1	80
35	IRS Tilt and Vertical Acceleration Error State Estimates for Flight #1	81
36	IRS Gyro Drift Error State Estimates for Flight #1	81
37	IRS Accelerometer and Barometric Altimeter Error State Estimates for Flight #1	82
38	Loran-C Position and Doppler Sea Bias Error State Estimates for Flight #1	82
39	Doppler Velocity Error State Estimates for Flight #1	83
40	Kalman-Corrected IRS Position Errors at VOT's for Flight #1	83
40a	Innovations for Position Measurements from Flight #1	84
40b	Innovations for Velocity Measurements from Flight #1	85
41	Loran-C Position Measurements for Flight #2	86
42	Doppler Velocity Measurements for Flight #2	87
43	IRS Position Errors at VOT's for Flight #2	88
44	Loran-C Position Errors at VOT's for Flight #2	88
45	IRS Position Error State Estimates for Flight #2	89
46	IRS Velocity Error State Estimates for Flight #2	89
47	IRS Tilt and Vertical Acceleration Error State Estimates for Flight #2	90
48	IRS Gyro Drift Error State Estimates for Flight #2	90
49	IRS Accelerometer and Baro Altimeter Error State Estimates for Flight #2	91
50	Loran-C Position Error State Estimates for Flight #2	91
51	Doppler Velocity Error State Estimates for Flight #2	92
52	Kalman-Corrected IRS Position Errors at VOT's for Flight #2	92
53	Comparison of Kalman Filter Configurations - Latitude Errors for Flight #1	93
54	Comparison of Kalman Filter Configurations - Longitude Errors for Flight #1	94

Figure		Page
55	Comparison of Kalman Filter Configurations - Latitude Errors for Flight #2	95
56	Comparison of Kalman Filter Configurations - Longitude Errors for Flight #2	96
57	Comparison of Kalman Filter Configurations - North Velocity Errors for Flight #1	97
58	Comparison of Kalman Filter Configurations - East Velocity Errors for Flight #1	98
59	Comparison of Kalman Filter Configurations - North Velocity Errors for Flight #2	99
60	Comparison of Kalman Filter Configurations - East Velocity Errors for Flight #2	100

SYMBOLS

Symbol	Definition
$B\omega_x$, $B\omega_y$, $B\omega_z$	Kalman filter Markov error states for IRS X, Y and Z axis gyro biases
Ba_x, Ba_y, Ba_z	Kalman filter Markov error states for IRS X, Y and Z accelerometer biases
Bh_b	Kalman filter Markov error state for baro-altimeter bias
B_{LAT}, B_{LNG}	Kalman filter Markov error states for Loran-C latitude and longitude offsets
B_V, B_W	Kalman filter Markov error states for Doppler V and W channel boresight errors
$\mathbf{C}_b^G$	Direction Cosine Matrix (DCM) defining the axis transformation from aircraft body axes (x, y, z) to local-level geographic axes (N, E, z)
$\mathbf{C}_b^{G^*}$	error-corrupted DCM, based on raw IRS Euler angle information
$\mathbf{C}_{bC}{}^G$	error-corrected DCM, based on knowledge of IRS tilt errors
$\mathbf{C}^b{}_G$	matrix defining the transformation from local-level geographic coordinates to body axis coordinates
$\mathbf{F}(t)$	error dynamics (or fundamental) matrix of the continuous-time version of the Kalman filter error state dynamics equation
$\mathbf{G}(t)$, $\mathbf{G}(k)$	continuous-time and discrete-time equivalent noise gain matrices of the Kalman filter error state dynamics equation
$\mathbf{H}(k)$	discrete-time observation matrix of the Kalman filter measurement equation at t_k
$\mathbf{H}_i(k)$	i^{th} row of $\mathbf{H}(k)$, associated with the i^{th} Kalman filter scalar measurement at t_k
$\mathbf{H}_{av}(k)$	averaged observation matrix used with prefiltering at t_k
$\mathbf{K}(k)$	Kalman gain matrix at filter discrete update time t_k
l_{DI}	lever arm position vector from Doppler antenna to LTN-90-100 IRS
l_x, l_y, l_z	three components of l_{DI} in the body axis frame
$\mathbf{P}(k)$	error covariance matrix associated with state vector $\mathbf{x}(k)$ or error state vector $\delta\mathbf{x}(k)$ at t_k
$\mathbf{Q}(k)$	covariance matrix associated with ZMWG discrete plant noise vector $\mathbf{u}(k)$ at t_k
$\mathbf{R}(k)$	covariance matrix associated with ZMWG discrete measurement noise vector $\mathbf{v}(k)$ at t_k

Symbol	Definition
$\mathbf{R}_{av}(k)$	averaged measurement noise covariance matrix used with prefiltering at t_k
$r_1,...,r_5$	noise variances associated with the individual scalar measurements
SF_U	Kalman filter Markov error state for Doppler U channel scale factor error
SB_N, SB_E	Kalman filter Markov error states for north and east sea bias components
U, **D**	Bierman's matrix factors of **P**, where $\mathbf{P} = \mathbf{UDU}^T$; **U** is upper triangular with unit diagonal and **D** is diagonal
U, V, W	general aircraft body axis velocity components
U_D, V_D, W_D	Doppler body axis velocity components
U_{DC}, V_{DC}, W_{DC}	Doppler body axis velocity components corrected for lever arm effects
U_I, V_I, W_I	IRS body axis velocity components
$\mathbf{u}(k)$	Kalman filter discrete ZMWG plant noise vector at t_k
$u\omega_x$, $u\omega_y$, $u\omega_z$	discrete-equivalent random drift in IRS X, Y and Z gyros
ua_x, ua_y, ua_z	discrete-equivalent random noise in IRS X, Y and Z accelerometers
ug_N, ug_E, ug_z	discrete-equivalent random gravity noise in N, E and z directions
uh_b	discrete-equivalent random noise in baro-altimeter
$uB\omega_x$, $uB\omega_y$, $uB\omega_z$	X, Y and Z components of plant noise for gyro Markov processes
uBa_x, uBa_y, uBa_z	X, Y and Z components of plant noise for accelerometer Markov processes
uBh_b	plant noise for baro-altimeter Markov process
uB_{LT}, uB_{LG}	latitude and longitude components of plant noise for Loran-C Markov processes
uSF_U, uB_V, uB_W	plant noise components for Doppler Markov processes
uSB_N, uSB_E	plant noise components for Doppler sea bias Markov processes
$\mathbf{v}(k)$	Kalman filter measurement noise vector at t_k
$\mathbf{v}_{av}(k)$	averaged measurement noise vector for prefiltering at t_k
$\mathbf{x}(k)$	state vector for discrete-equivalent dynamic model at t_k

Symbol	Definition
$\mathbf{z}(k)$	Kalman filter measurement vector at t_k
$\mathbf{z}_{av}(k)$	averaged measurement vector for prefiltering at t_k
$z_1,\ldots,z_5$	individual scalar measurements that comprise $\mathbf{z}(k)$
$\delta\mathbf{x}(k)$	Kalman filter error state vector for error dynamics model at t_k
δL, $\delta\lambda$, δh	Kalman filter error states for the 3 IRS position errors
δv_N, δv_E, δv_z	Kalman filter error states for the 3 IRS velocity errors
δa	Kalman filter error state for vertical acceleration error
$\delta\mathbf{C}_b^G$	error matrix associated with error-corrupted DCM $\mathbf{C}_b^{G^*}$
δt_i	incremental time interval within basic Kalman filter update interval ΔT
ΔT	time interval between Kalman filter updates
$\boldsymbol{\varepsilon}$	tilt error vector depicting attitude errors in IRS computed geographic frame relative to the true geographic frame
ε_N, ε_E, ε_z	Kalman filter error states for the 3 IRS tilt errors
$\boldsymbol{\varepsilon}\times$	matrix-equivalent of the skew symmetric cross product operator associated with $\boldsymbol{\varepsilon}$
$\boldsymbol{\Phi}(k,k+1)$	state transition matrix over the time interval t_k --> t_{k+1}
$\nu_i(k)$, $p\{\nu_i(k)\}$	innovations process and associated variance for i^{th} scalar measurement at t_k
θ, ϕ, ψ	aircraft Euler angles: pitch, roll and heading
θ_C, ϕ_C, ψ_C	Kalman filter-corrected Euler angles
τ_i	correlation time for the i^{th} Kalman filter error state Markov process

A KALMAN FILTER INTEGRATED NAVIGATION DESIGN FOR THE IAR TWIN OTTER ATMOSPHERIC RESEARCH AIRCRAFT

1.0 INTRODUCTION

The IAR Twin Otter Atmospheric Research Aircraft has a continuing requirement for more accurate, inertially-based navigation data for both track recovery and the calculation of wind gust components. This navigational accuracy is necessary, not just during post-flight analysis, but also for real-time, in-flight guidance and wind computation. Previous developmental work at the Flight Research Laboratory (FRL) on advanced navigation systems has demonstrated the benefits of a Kalman filter integrated navigation approach in satisfying the most stringent navigational requirements (Refs. 1, 2, 3). A significant upgrade to the navigation sensor suite onboard the Twin Otter in the last two years has resulted in the potential, via Kalman filtering, for generating very high quality inertial velocity and positional information in real time, together with improved airborne wind components. Figure 1 shows the IAR Twin Otter's suite of navigation, inertial sensing and air data instrumentation.

The Kalman filter integrated navigation design described in this report is based on the optimal blending of data from an LTN-90-100 strapdown Inertial Reference System (IRS), a Decca Type 72 Doppler velocity sensing (DVS) system and an ARNAV R-40 airborne Loran-C receiver - sensors that are available on the Twin Otter at the present time. In the Twin Otter's real-time computing/data acquisition system, all three of these navigation sensors are interfaced to the onboard LSI-11/73 microcomputer system, and a complete set of navigation parameters is being recorded. In particular, all of the raw inertial data parameters from the LTN-90-100 IRS, required for proper design of an IRS-based Kalman filter, are available with sufficient resolution and at a suitable digital sampling rate.

A driving force behind the decision to pursue this integrated navigation approach on the Twin Otter has been the observation that significant velocity errors (and, eventually, position errors) can occur in the LTN-90-100 IRS over the course of a flight, and the observed error levels can seriously degrade the accuracy of the wind calculations. On the other hand, the airborne Loran-C positional data has been demonstrated to be consistently more accurate than the IRS position information, in the long term. An integrated navigation system approach, using the principles of Kalman filtering, has the ability to use Loran-C data (and, to a lesser extent, Doppler velocity data) to accurately track the dominant IRS errors (in both the position and velocity components), and provide IRS error corrections at a rate appropriate for Twin Otter requirements.

In this report, Section 2 gives a short description, in general terms, of the mathematical nomenclature and the fundamental equations used to formulate a linear, discrete-time Kalman filter. As well, computationally efficient subroutines for performing linear, discrete-time Kalman filtering are described. Finally, the application of Kalman filtering to the general problem of integrated airborne navigation is introduced. Section 3, along with an associated set of four appendices, then gives complete details concerning the specific Twin Otter IRS/Doppler/Loran-C Kalman filter design that is the main subject of this report.

A comprehensive set of software for simulating various navigation sensors has been developed at the FRL to accurately model realistic aircraft flight trajectory dynamics and error dynamics in sensors such as medium-accuracy INS's, Doppler velocity sensing (DVS) systems and various radio position fixing systems. Section 4 describes the various software modules that make up this complete simulation package, as well as the software for performing the actual integrated navigation Kalman filtering. An extensive series of Kalman filter runs using this simulation software has demonstrated the fundamental robustness and accuracy of the Kalman filter design. The simulation results indicate that there is an incremental improvement to be gained by using measurement prefiltering on both the Loran-C and Doppler-based measurements, but the improvement is not considered significant enough to warrant

using prefiltering in a real-time application. Simulation runs also suggest that, in the presence of the reasonably accurate Loran-C position-based measurements, the rather noisy and inaccurate Doppler velocity measurements contribute very little to the estimation of the various IRS error states. Filter robustness checks have confirmed that there is virtually no difference in performance when using an 8 Hz version of the IRS digital data rather than the usual 16 Hz version, or when updating the Kalman filter every 20 seconds rather than every 10 seconds.

Section 5 describes, in some detail, Kalman filter results that are based on two sets of real flight test data acquired specifically to assess filter performance under typical aircraft operational conditions. Most of the results and conclusions established from the simulation studies have been corroborated from an analysis of the real flight data. In particular, it has been confirmed that there is, indeed, no further improvement in IRS error estimation accuracy to be gained by processing the Doppler-based measurements along with the Loran-C measurements. In fact, one must be careful to establish the proper measurement noise 'weighting' of the Doppler-based data so as to avoid having it actually degrade filter performance when used in conjunction with the Loran-C data. The best one can achieve is an effective ongoing calibration of the Doppler velocity sensing system which would be useful in a situation where either the IRS or Loran-C sensor failed in flight. From the analysis of real flight data, it has been found that under certain circumstances, the use of Visual On-Tops (VOT's) as extra measurements can cause an undesirable transient in several of the Kalman filter error states. The procedures developed for using VOT's as measurements must be checked out very carefully, for any given Kalman filter configuration, to ensure that basic filtering integrity is maintained.

General conclusions and future work are discussed in Section 6. The application of Kalman filtering techniques to the analysis of Twin Otter flight data is an evolving process with several hardware and software changes planned for the near future as outlined in Section 6.

2.0 THE KALMAN FILTER EQUATIONS FOR INTEGRATED NAVIGATION

2.1 The Linear Discrete-Time Kalman Filter

For completeness, and to establish the nomenclature that will be used throughout the report, the general form of the linear, discrete-time version of the Kalman filter equations will be outlined here. Readers are directed to Refs. 4, 5 and 6 for a thorough discussion of the fundamental theory of Kalman filtering.

Assume that a physical system has an equivalent n^{th}-order, discrete-time dynamic model of the form

$$\mathbf{x}(k+1) = \mathbf{\Phi}(k,k+1)\,\mathbf{x}(k) + \mathbf{G}(k)\,\mathbf{u}(k) \qquad (1)$$

where $\mathbf{x}(k)$ is the n^{th}-order system state vector evaluated at discrete time t_k; $\mathbf{\Phi}(k,k+1)$ is the n x n state transition matrix over the time interval t_k --> t_{k+1}; $\mathbf{G}(k)$ is the n x r plant noise gain matrix at t_k; and $\mathbf{u}(k)$ is the r^{th}-order vector of zero-mean, white, Gaussian (ZMWG) discrete plant noise processes having covariance matrix $\mathbf{Q}(k)$ at t_k.

For the above dynamic system, let a discrete-time, m^{th}-order measurement process exist in the form

$$\mathbf{z}(k+1) = \mathbf{H}(k+1)\,\mathbf{x}(k+1) + \mathbf{v}(k+1) \qquad (2)$$

where $\mathbf{z}(k+1)$ is the measurement vector at time t_{k+1}, $\mathbf{H}(k+1)$ is the m x n observation matrix and $\mathbf{v}(k+1)$ is the m^{th}-order measurement noise vector having covariance matrix $\mathbf{R}(k+1)$ at t_{k+1}. Assume that noise vectors $\mathbf{u}$ and $\mathbf{v}$ are statistically independent (i.e. the components of $\mathbf{u}$ are uncorrelated with the components of $\mathbf{v}$); and assume also that $\mathbf{x}(0)$ is independent of both $\mathbf{u}$ and $\mathbf{v}$. Define the following vector and matrix variables:

$\mathbf{x}'(k+1)$, $\mathbf{P}'(k+1)$ - time update of the state vector and its covariance at t_{k+1} (i.e. just prior to a measurement update at time t_{k+1}).

$\mathbf{x}(k+1)$, $\mathbf{P}(k+1)$ - optimal state estimate and its associated covariance at t_{k+1} (i.e. just after a measurement update at time t_{k+1}).

$\mathbf{x}_0$, $\mathbf{P}_0$ - initial conditions on the state vector and its covariance (i.e. $\mathbf{x}(0) = \mathbf{x}_0$, $\mathbf{P}(0) = \mathbf{P}_0$).

Under the foregoing definitions and assumptions, it can be shown that the optimal estimate of the state vector at time t_{k+1} (i.e. $\mathbf{x}(k+1)$) and its associated error covariance (i.e. $\mathbf{P}(k+1)$) can be computed from the following set of five recursion equations that form the heart of discrete-time Kalman filtering:

$$\mathbf{x}'(k+1) = \Phi(k,k+1)\,\mathbf{x}(k) \qquad \text{state time update}$$

$$\mathbf{P}'(k+1) = \Phi(k,k+1)\,\mathbf{P}(k)\,\Phi^T(k,k+1) + \mathbf{G}(k)\,\mathbf{Q}(k)\,\mathbf{G}^T(k) \qquad \text{error covariance time update}$$

$$\mathbf{K}(k+1) = \mathbf{P}'(k+1)\,\mathbf{H}(k+1)^T\,[\mathbf{H}(k+1)\,\mathbf{P}'(k+1)\,\mathbf{H}^T(k+1) + \mathbf{R}(k+1)]^{-1} \qquad \text{Kalman gain}$$

$$\mathbf{x}(k+1) = \mathbf{x}'(k+1) + \mathbf{K}(k+1)\,[\mathbf{z}(k+1) - \mathbf{H}(k+1)\,\mathbf{x}'(k+1)] \qquad \text{state measurement update}$$

$$\mathbf{P}(k+1) = \mathbf{P}'(k+1) - \mathbf{K}(k+1)\,\mathbf{H}(k+1)\,\mathbf{P}'(k+1) \qquad \text{error covariance measurement update} \qquad (3)$$

with initialization of this recursive procedure provided by a priori knowledge of $\mathbf{x}_0$ and $\mathbf{P}_0$. It should be noted that, when an error state Kalman filter is being used (see subsection 2.3), the state vector in Eqns. 3 is denoted as $\delta\mathbf{x}$.

2.2 Software for Performing Discrete-Time Kalman Filtering

Numerically efficient software exists for performing the foregoing set of five recursion equations, especially if certain simplifying assumptions can be made. This author uses a set of four relatively short subroutines to perform the above vector/matrix calculations in a very efficient fashion. They are based on Bierman's $\mathbf{UDU}^T$ factorization algorithms, which are generally acknowledged to be the most numerically stable, computationally efficient ones to use, especially for real-time applications. In order to avoid the matrix inversion that would normally be required in the Kalman gain equation of Eqns. 3, the assumption is made that the elements of $\mathbf{z}$, the measurement vector, are statistically independent of each other. This assumption is certainly valid for the kinds of measurements that are described in subsection 3.3. Under this assumption, the covariance matrix $\mathbf{R}$ is diagonal and it can be shown (Ref. 7) that the measurement vector $\mathbf{z}$ can then be processed in a one-at-a-time fashion. In essence, the Kalman gain vector/matrix equation is converted to a sequence of much simpler scalar/vector operations. The four subroutines that make up the $\mathbf{UDU}^T$ factorization version of Kalman filtering are described as follows:

i) **SUBROUTINE FACTOR(N,NN,P,U)** - a subroutine to factor the covariance matrix **P** into matrices **U** and **D**, where $\mathbf{P} = \mathbf{UDU}^T$. **U** is upper triangular with unit diagonal, and **D** is a diagonal matrix. The inputs to the subroutine are as follows:

N - dimension of the state vector **x**;

NN - state dimension augmented by plant noise dimension
i.e. NN = N + NW, where NW is size of **u**;

P(N,N) - covariance matrix to be factored.

The outputs from the subroutine are the following:

U(N,NN) - the matrix of the factors of **P**, with **D** on the diagonal and extra storage for matrix **G**;

P(N,N) - modified covariance matrix, with upper triangular portion overwritten.

ii) **SUBROUTINE UNFACT(N,NN,P,U)** - this subroutine reconstructs the **P** matrix from its $\mathbf{UDU}^T$ decomposition. The inputs to the subroutine are the following:

N - dimension of the state vector **x**;

NN - state dimension augmented by plant noise dimension;

U(N,NN) - the matrix of the factors of **P**, with **D** on the diagonal and extra storage for **G**.

The outputs from the subroutine are:

P(N,N) - reconstructed covariance matrix.

iii) **SUBROUTINE MWGSUD (N,NW,NN,U,G,PHI,Q,X,A,V,D)** - this subroutine performs Bierman's modified weighted Gram-Schmidt (MWGS) time update of state estimate **x** and its factored covariance matrix **U** (i.e. the first two equations in the five-equation set of Eqns. 3). The inputs to this subroutine are:

N - dimension of the state vector **x**;

NW - dimension of the plant noise vector **u**;

NN - augmented state dimension, where NN = N + NW;

U(N,NN) - a matrix that contains the factored covariance **U**(N,N) in the first N columns and **G**(N,NW) in the remaining NW columns;

G(N,NW) - plant (process) noise gain matrix (**G** in Eqn. 1);

PHI(N,N) - linear state transition matrix (Φ in Eqn. 1);

Q(NW) - vector of plant noise covariance diagonal elements;

X(N) - optimal state estimate from previous Kalman filter iteration.

The outputs from this subroutine are the following:

X(N) - time updated state estimate (first of Eqns. 3);

U(N,NN) - time updated covariance factors (second of Eqns. 3);

A(NN), V(NN), D(NN) - double precision working vectors.

iv) **SUBROUTINE UDUPDC(N,NN,X,U,ZZ,RR,H,K,RSD,ALPHA,ITST,IPASS,A,B)** - this subroutine performs Bierman's $\mathbf{UDU}^T$ measurement update for a scalar measurement z (i.e. the last three of Eqns. 3). Prior to updating **U** and **x**, a tolerance test can be performed on the measurement residual. The measurement residuals sequence (also called the innovations) and an estimate of its variance are computed as well. If the residual lies outside of the 1-sigma boundaries defined by the tolerance, the subroutine returns without updating **U** and **x**. The inputs to this subroutine are:

N - dimension of state vector **x**;

NN - state dimension augmented by plant noise dimension;

X(N) - state vector from a time update or intermediate measurement update;

U(N,NN) - factored covariance matrix from a time update or intermediate measurement update;

ZZ - scalar measurement z to be processed;

RR - noise variance associated with the measurement;

H(N) - geometry vector for the measurement z (i.e. appropriate row of observation matrix **H**);

ITST - integer multiplier to set tolerance for residuals monitoring. If ITST is set negative, then residuals monitoring is omitted.

The outputs from the subroutine are as follows:

X(N) - updated state vector based on the measurement;

U(N,NN) - updated covariance matrix in factorized form;

K(N) - Kalman gain vector associated with the measurement (i.e. appropriate column of Kalman gain matrix **K** in the third one of Eqns. 3);

RSD - latest value of measurement residuals (innovation);

ALPHA - latest estimate of the variance of the residuals sequence;

IPASS - integer flag to indicate if measurement is used (IPASS = 1) or not used (IPASS = 0) when residuals monitoring is invoked;

A(N), B(N) - working vectors.

2.3 Integrated Navigation Using Kalman Filtering

There are three generally accepted configurations for the use of Kalman filtering to blend, in an optimal fashion, navigation data from various airborne transducers. These three configurations (Refs. 3, 8) are known as the i) total state representation, the ii) error state feedforward representation and the iii) error state feedback representation. In all three cases, navigation data from a dead reckoning (DR) navigator such as inertial or Doppler/heading reference are blended together with redundant navigation data from one or more external navaids such as Omega/VLF, airborne Loran-C, Navstar GPS or VOR/DME. The optimal blending is accomplished in real time through the use of a Kalman filter algorithm that differs somewhat depending upon the configuration being used.

In this report, the error state feedforward configuration will be used exclusively. Figure 2 shows a simplified block diagram representation of this configuration. In the error state feedforward implementation, the inertial navigation system (INS) or Doppler/heading dead reckoning (DR) system is treated as a 'black box' that outputs position, velocity and (possibly) attitude data corrupted with errors. Measurements, in the form of position, velocity or range differences between DR predictions and what is actually being sensed by an external navaid, are sent to the Kalman filter along with fundamental data generated by the DR device. The Kalman filter then estimates the dominant, low frequency error states of the DR system and the external navaids in real time (i.e. as they are occurring). The DR errors are nulled out, in a feedforward sense, to produce best estimates of navigation variables such as 3-D position, velocity and the Euler angles (i.e. θ, ϕ and ψ).

It is worthwhile to list the salient features of the error state feedforward approach to Kalman filter integrated navigation:

- Any potentially nonlinear navigation calculations in the DR 'black box' are performed separately from the Kalman filter recursive update calculations.

- Any required nonlinear pre-processing of external navaid data also occurs separately from the Kalman filter equations. Position, velocity and/or range differences between DR predictions and navaid outputs are calculated and then sent to the Kalman filter as measurement data.

- Under the foregoing conditions, the Kalman filter can be effectively formulated as a sub-optimal, reduced-order, linear Kalman filter (Ref. 8) which will estimate the dominant navigation error states, as opposed to the actual physical states of the navigation system. Also, by using the error state formulation, the use of a linear Kalman filter design is more likely to be a valid assumption.

- The error states and the measurements used in the Kalman filter are all relatively small in magnitude, so numerical round-off error is usually not a problem even when cycling through the recursion equations (i.e. Eqns. 3) many times.

- The dominant error states of any navigation sensor vary slowly in time relative to the actual sample rate of the navigation data. Hence, discrete-time Kalman filtering can take place at a slow update rate; but the error states so estimated can be applied to the navigation data at the much faster navigation data rate.

- With a discrete-time, linear Kalman filter design, efficient 'off-the-shelf' FORTRAN software, such as Bierman's $\mathbf{UDU}^T$ factorization algorithms described in subsection 2.2, can be used (Ref. 7).

3.0 IRS/DOPPLER/LORAN-C KALMAN FILTER NAVIGATOR

3.1 Navigation Sensor Descriptions

At the present time, the Twin Otter's suite of navigation sensors (Fig. 1) available for optimal blending is as follows:

i) **Litton LTN-90-100 Inertial Reference System** - this modern, three-axis strapdown inertial system (called an IRS - Inertial Reference System - by Litton) utilizes ultra-reliable ring laser gyro technology and a digital data bus (ARINC 429 standard) to provide a complete set of inertial parameters in digital form, many at a 64 Hz datarate. Table 3.1 gives a list of the 16 inertial parameters recorded from the digital data bus (and required for the IRS-based Kalman filter design) together with the update rate, units, dynamic range, resolution and positive sense for each parameter. The specific interface design for the Twin Otter onboard data acquisition system is such that all 16 IRS parameters are sampled at 16 Hz. Table 3.2 lists the fundamental specifications of the system's accelerometers and gyroscopes - these statistical specifications will be of importance in the Kalman filter design. This particular IRS requires barometric altimeter data, in ARINC 429 format, as an input in order to stabilize the vertical channel via an internal, third-order, fixed-gain digital filter. Typical accuracy for this type of IRS would be 1 nm/hr in the horizontal position components, 5 knots in the horizontal velocity components, 0.05 deg for the pitch and roll attitude components and 0.2 deg for the heading. The reader is referred to Litton's Technical Description Manual (Ref. 9) for the LTN-90-100 IRS for further technical details about this IRS.

ii) **Decca Doppler Radar Type 72** - this 3-beam Janus Doppler radar system measures the three body-axis (i.e. strapdown) components of aircraft velocity, namely U, V and W. The specially-designed digital interface for this unit is such that each Doppler velocity component is averaged over a 1/2 second interval and then sampled, thus yielding an effective sample rate of 2 Hz. Table 3.3 shows some of the important characteristics of this Doppler radar system as it is interfaced to the Twin Otter onboard data acquisition system, including the range, accuracy and resolution of each of the velocity components. For further information on the technical details of this Doppler system, the reader should consult Ref. 10.

iii) **ARNAV R-40-AVA-1000A Loran-C Receiver** - this airborne Loran-C receiver provides digitized geographical latitude and longitude data, at a nominal 1 Hz update rate, via a standard RS-232C serial output port. Within the normal coverage area of one of the Loran-C station chains, expected accuracy is 0.2 nm or better in each of latitude and longitude. The resolution of the Loran-C latitude and longitude data, as digitized onboard the Twin Otter, is 0.01 arc minutes. The Operation Manual for the R-40 Loran-C receiver (Ref. 11) provides technical information on the Loran-C system as a whole, as well as on the R-40 receiver itself. As an indication of expected airborne Loran-C performance, Ref. 12 shows results of flight tests that took place during the certification of an airborne Loran-C navigation system.

3.2 Overview of the Kalman Filter Navigator

A simple block diagram representation of the proposed IRS/Doppler/Loran-C Kalman filter navigator is shown in Fig. 3. This figure shows a specific implementation of the error state feedforward version of a Kalman filter integrated navigation scheme. In this particular design, the LTN-90-100 IRS is the dead reckoning (DR) system that outputs error-corrupted position, velocity and attitude information at a 16 Hz datarate. Also shown is the air data source of altitude information that is required by the IRS for its third-order baro damping loop (see Appendix A for loop details). Two other navaids, the airborne Loran-C receiver and the Doppler radar system, supply redundant navigation information that is used to form measurements to be processed by the error state Kalman filter. In the filter design shown here, the differences in position (both latitude and longitude components) between the IRS and the Loran-C receiver are used as measurements for the Kalman filter. This particular measurement type could be processed by the Kalman filter as frequently as 1 Hz, since that is the rate at which Loran-C data is sampled.

TABLE 3.1

DIGITAL DATA PARAMETERS AVAILABLE FROM LTN-90-100 IRS

	PARAMETER	RATE (HZ)	UNITS	RANGE(+/-)	RESOLUTION	+'VE SENSE
1.	Latitude	8	Degs	180	.000172	North Fr. 0°
2.	Longitude	8	Degs	180	.000172	East Fr. 0°
3.	True Heading	32	Degs	180	.005493	CW Fr. North
4.	Pitch Angle	64	Degs	180	.005493	Up
5.	Roll Angle	64	Degs	180	.005493	Rgt Wing Down
6.	Body Ptch Rate	64	Deg/S	128	.003906	Up
7.	Body Roll Rate	64	Deg/S	128	.003906	Rgt Wing Down
8.	Body Yaw Rate	64	Deg/S	128	.003906	Nose Right
9.	Body Long Acc	64	G	4	.000122	Forward
10.	Body Lat Accel	64	G	4	.000122	Right
11.	Body Norm Acc	64	G	4	.000122	Up
12.	Inertial Alt'd	32	Feet	131,072	0.125	Up
13.	Vertical Accel	64	G	4	.000122	Up
14.	Inert Vert Spd	32	Ft/Min	32,768	1.00	Up
15.	N-S Velocity	16	Knots	4096	0.125	North
16.	E-W Velocity	16	Knots	4096	0.125	East

TABLE 3.2

SPECIFICATIONS FOR IRS ACCELEROMETERS AND GYROSCOPES

A-4 ACCELEROMETERS:

Scale Factor Repeatability	0.005% (1-sigma,1 year)
Scale Factor Nonlinearity	10 μg/g ^{2} (1-sigma)
Bias Repeatability	50 μg (1-sigma, 1 year)
Bias Short Term Stability	5 μg (1-sigma)
Alignment Stability	5 arc sec (1-sigma)
Maximum Acceleration	25 g

LG-8028B GYROS:

Scale Factor Repeatability	0.0005% (1-sigma)
Bias Repeatability	0.01 deg/hr (1-sigma)
Random Drift	0.003 deg/√hr (1-sigma)
Maximum Rate	400 deg/s

TABLE 3.3

DIGITAL DATA PARAMETERS AVAILABLE FROM DECCA DOPPLER SYSTEM

COMPONENT	UNITS	RANGE	RESOLUTION	ACCURACY	+'V SENSE
Long Vel (U)	Knots	497.3	0.0194	2.0	Forward
Lat Vel (V)	Knots	248.6	0.0097	2.0	Right
Norm Vel (W)	Knots	124.3	0.0039	2.0	Down

The second type of measurement data available for processing is derived from the Doppler strapdown velocity components. As can be seen in Fig. 3, Doppler U,V,W velocity components are differenced with their IRS counterparts, and these strapdown (i.e. body axis) velocity differences are then processed by the Kalman filter. The data rate for velocity measurements is the fundamental Doppler rate of 2 Hz. In the error state Kalman filter design, dominant sources of error in the IRS, Loran-C receiver and Doppler radar system are modelled and estimated in an on-line fashion as the measurement data are processed. A simple corrector algorithm is then used to transform tilt angle error estimates into Euler angle errors, and the 16 Hz data stream of raw IRS data (position, velocity and attitude components) is corrected, in real time, using the slowly varying error estimates generated by the Kalman filter. The basic update interval for the Kalman filter processing (i.e. executing Eqns. 3 for a new set of measurement data) is set at ten seconds - more than adequate for 'tracking' the expected sources of error in the various navigation systems. Not only is the IRS corrected in a feedforward sense - the Loran-C receiver and Doppler radar system are corrected for their dominant low frequency errors as well. This on-line calibration of the redundant navaids, as well as the IRS, allows for the possibility of a 'graceful degradation' in navigational accuracy should the main IRS-based DR system fail in flight.

3.3 Details of the Kalman Filter Design

3.3.1 Error States Chosen

For the complete IRS/Doppler/Loran-C Kalman filter, there are a total of 24 error states modelled (Appendix B shows details of the error modelling for the IRS, Doppler and Loran-C systems). The error states are divided into two groups - i) 10 system error states that relate to the basic baro-damped IRS (i.e. errors in inertial position, velocity, attitude components and vertical loop acceleration error); and ii) 14 first-order Markov error states that correspond to the slowly varying, bias-like errors assumed to exist in the inertial sensors (i.e. bias errors in accelerometers, gyros, altimeter) and redundant navaids (i.e. Loran-C offsets, Doppler scale factor and boresight errors, sea currents). Table 3.4 lists all of the system error states, together with typical values of the 1-sigma levels that would be used for the initial conditions, P_0, when running the Kalman filter. Table 3.5 is a list of the 14 Markov error states, again showing typical 1-sigma, initial condition values as well as the nominal values of correlation times that would be assumed for the associated first-order Markov error processes. It should be noted that the last two error states in Table 3.5, the two sea bias components, would only be included in the Kalman filter in the case of an overwater flight. Note, also, that in these tables the engineering units shown correspond to those being used internally by the Kalman filter and are <u>not</u> metric units, as it turns out.

3.3.2 Plant Dynamics (Φ, G, Q)

The full 24 x 24 discrete state transition matrix **Φ** for the Kalman filter design is derived from the continuous-time version of the error state equations, the details of which have been developed in Appendix B. The general relationship between the continuous-time matrices, **F**(t) and **G**(t), (shown in Eqn. B1 of Appendix B) and their discrete-equivalent counterparts, **Φ**(k,k+1) and **G**(k), can be expressed as follows:

$$\Phi_{i,j}(k,k+1) = \int_{t_k}^{t_{k+1}} F_{i,j}(t)dt \; ; \; i,j = 1,...,24; \; i \neq j$$

$$\Phi_{i,i}(k,k+1) = 1 + \int_{t_k}^{t_{k+1}} F_{i,i}(t)dt \; ; \; i = 1,...,10$$

$$\Phi_{i,i}(k,k+1) = \exp(-\Delta T/\tau_i) \; ; \; i = 11,...,24$$

$$G_{i,j}(k) = (1/\Delta T) \int_{t_k}^{t_{k+1}} G_{i,j}(t)dt \; ; \; i,j = 1,...,24 \qquad (4)$$

where $\Delta T = t_{k+1} - t_k$ is the Kalman filter update interval of ten seconds, the τ_i's are the Markov error state correlation times and the integrations are performed numerically using a simple trapezoidal integration algorithm.

TABLE 3.4

IRS SYSTEM ERROR STATES AND STATISTICS

ERROR STATE	DESCRIPTION	INITIAL RMS VALUE	UNITS
1. δL	Latitude Error	2.91×10^{-5}	Rad
2. $\delta\lambda$	Longitude Error	4.11×10^{-5}	Rad
3. δh	Altitude Error	30	Feet
4. δv_N	North Velocity Error	0.33	Ft/Sec
5. δv_E	East Velocity Error	0.33	Ft/Sec
6. δv_z	Vertical Velocity Error	0.33	Ft/Sec
7. ε_N	N Axis Attitude Error	5.0×10^{-5}	Rad
8. ε_E	E Axis Attitude Error	5.0×10^{-5}	Rad
9. ε_z	z Axis Attitude Error	1.5×10^{-3}	Rad
10. δa	V Accel Correction	0.25×10^{-1}	Ft/S^2

TABLE 3.5

MARKOV ERROR STATES AND STATISTICS

ERROR STATE	DESCRIPTION	INITIAL RMS VALUE	UNITS	CORREL. TIME
11. $B\omega_x$	X Axis Gyro Bias	9.0×10^{-8}	Rad/S	7,500 S
12. $B\omega_y$	Y Axis Gyro Bias	9.0×10^{-8}	Rad/S	7,500 S
13. $B\omega_z$	Z Axis Gyro Bias	9.0×10^{-8}	Rad/S	7,500 S
14. Ba_x	X Axis Accel Bias	2.0×10^{-3}	Ft/S^2	15,000 S
15. Ba_y	Y Axis Accel Bias	2.0×10^{-3}	Ft/S^2	15,000 S
16. Ba_z	Z Axis Accel Bias	2.0×10^{-3}	Ft/S^2	15,000 S
17. Bh_b	Baro Altimeter Bias	600	Feet	5,000 S
18. B_{LAT}	Loran Latitude Bias	8.73×10^{-5}	Rad	15,000 S
19. B_{LNG}	Loran Longitude Bias	1.23×10^{-4}	Rad	15,000 S
20. SF_U	Doppler SF Error	2 %	----	15,000 S
21. B_V	V Dop Boresight Error	0.02	Rad	15,000 S
22. B_W	W Dop Boresight Error	0.02	Rad	15,000 S
23. SB_N	N Sea Bias Component	4.5	Ft/S	900 S
24. SB_E	E Sea Bias Component	4.5	Ft/S	900 S

A complete list of the discrete-equivalent ZMWG noise components included in the Kalman filter design is shown in Table 3.6, along with typical values for the associated standard deviations. Note that the first ten components in Table 3.6 correspond to integrated random noise effects over the Kalman filter update interval (i.e. ΔT) of ten seconds. The rest of the noise components in Table 3.6 are discrete-equivalent versions of the so-called Markov plant noises (Ref. 8) associated with error states #11 to #24 in Table 3.5. Their standard deviations are determined from the filter update interval (ΔT), plus the information given in Table 3.5 for the correlation time and initial RMS level of the associated Markov error state. For simplicity in the filter design, all plant noise components are assumed to have constant variances that are not affected by aircraft manoeuvring. This results in a plant noise covariance matrix **Q** that is diagonal and has all components constant with time.

3.3.3 Measurement Process (H, R)

Recall that there are two types of measurements to be processed by the Kalman filter, namely i) measurements based on Loran-C position data, and ii) measurements based on Doppler velocity data. Details of the mathematical developments for each type of measurement are shown in Appendix C, while only the important end results are given here.

Measurements Based On Loran-C -

These measurements, taken at discrete times t_k, will consist of the two simultaneous, independent difference quantities, $z_1(k)$ and $z_2(k)$, where

$$z_1(k) = LAT_{IRS}(t_k) - LAT_{LOR}(t_k)$$

$$z_2(k) = LNG_{IRS}(t_k) - LNG_{LOR}(t_k) \qquad (5)$$

TABLE 3.6

PLANT NOISE COMPONENTS AND STATISTICS

	COMPONENT	DESCRIPTION	RMS VALUE	UNITS
1.	$u\omega_x$	X Gyro Random Drift	4.0×10^{-8}	Rad
2.	$u\omega_y$	Y Gyro Random Drift	4.0×10^{-8}	Rad
3.	$u\omega_z$	Z Gyro Random Drift	4.0×10^{-8}	Rad
4.	ua_x	X Accel Random Noise	1.0×10^{-3}	Ft/S
5.	ua_y	Y Accel Random Noise	1.0×10^{-3}	Ft/S
6.	ua_z	Z Accel Random Noise	1.0×10^{-3}	Ft/S
7.	ug_N	N Random Gravity	1.0×10^{-3}	Ft/S
8.	ug_E	E Random Gravity	1.0×10^{-3}	Ft/S
9.	ug_z	Z Random Gravity	1.0×10^{-3}	Ft/S
10.	uh_b	Altimeter Random Noise	31.62	Ft-S
11.	$uB\omega_x$	X Gyro Markov Plant Noise	0.46×10^{-8}	Rad/S
12.	$uB\omega_y$	Y Gyro Markov Plant Noise	0.46×10^{-8}	Rad/S
13.	$uB\omega_z$	Z Gyro Markov Plant Noise	0.46×10^{-8}	Rad/S
14.	uBa_x	X Acc Markov Plant Noise	7.3×10^{-5}	Ft/S^2
15.	uBa_y	Y Acc Markov Plant Noise	7.3×10^{-5}	Ft/S^2
16.	uBa_z	Z Acc Markov Plant Noise	1.46×10^{-4}	Ft/S^2
17.	uBh_b	Altim Markov Plant Noise	37.91	Feet
18.	uB_{LT}	Loran Lat Markov Plant Noise	0.32×10^{-5}	Rad
19.	uB_{LG}	Loran Lng Markov Plant Noise	0.45×10^{-5}	Rad
20.	uSF_U	Dop SF Markov Plant Noise	0.73×10^{-3}	----
21.	uB_V	V Boresight Markov Noise	0.73×10^{-3}	Rad
22.	uB_W	W Boresight Markov Noise	0.73×10^{-3}	Rad
23.	uSB_N	N Sea Bias Markov Noise	0.667	Ft/S
24.	uSB_E	E Sea Bias Markov Noise	0.667	Ft/S

with LAT_{IRS}, LNG_{IRS} the IRS estimates of geographical position and LAT_{LOR}, LNG_{LOR} the Loran-C estimates of position. The 24-element rows of the **H** observation matrix that are associated with these two measurements (i.e. the first two rows of **H**) are given very simply as (see Appendix C for details):

$$\mathbf{H}_1 = (1\,0\,0\,0\,.\,.\,.\,.\,.-1\,0\,0\,0\,0\,0\,0)\ ;\ \text{'-1' is } 18^{th} \text{ element}$$

$$\mathbf{H}_2 = (0\,1\,0\,0\,.\,.\,.\,.\,.\,0-1\,0\,0\,0\,0\,0)\ ;\ \text{'-1' is } 19^{th} \text{ element} \qquad (6)$$

The noise variances assumed for these two measurements, r_1 and r_2, are the first two diagonal elements of the overall measurement noise covariance matrix **R**. Nominal values of $r_1 = (0.07 \text{ arc min})^2$ and $r_2 = (0.1 \text{ arc min})^2$ are assigned to these parameters, based on the observed performance of the Loran-C receiver onboard the IAR Twin Otter. The Loran-based measurement data are available at a 1 Hz rate; but the normal Kalman filter measurement update rate is only 0.1 Hz (i.e. every ten seconds).

Measurements Based On Doppler Radar -

Recall that the fundamental measurements produced by the Doppler radar system are the body axis velocity components U_D, V_D, W_D (forward, to the right and down being the positive senses in the body axis coordinate frame). It is then necessary to process velocity differences between the IRS and the Doppler system in this body axis frame. In order to do this properly, the IRS velocity components must be transformed into equivalent body axis components and, as well, the Doppler velocity components must be corrected for lever arm effects. The lever arm effects are due to the fact that the location of the Doppler radar antenna is not coincident with the location of the IRS.

Let $\mathbf{C}^b_G$ be the transformation matrix that converts IRS velocity data in geographic coordinates (i.e. v_N, v_E, v_z) into equivalent components in body axis coordinates (i.e. U_I, V_I, W_I). This transformation matrix is re-computed continuously, based on the Euler angle (i.e. attitude) data available from the LTN-90-100 IRS, and the individual matrix element calculations are shown in Appendix C. The IRS velocity components, converted into body axis coordinates, can then be computed as

$$\begin{aligned} U_I &= C_{1,1}\,v_N + C_{1,2}\,v_E + C_{1,3}\,v_z \\ V_I &= C_{2,1}\,v_N + C_{2,2}\,v_E + C_{2,3}\,v_z \\ W_I &= C_{3,1}\,v_N + C_{3,2}\,v_E + C_{3,3}\,v_z \end{aligned} \qquad (7)$$

In order to look at lever arm effects, define l_{DI} as the lever arm position vector <u>from</u> the Doppler radar antenna <u>to</u> the LTN-90 IRS. This position vector is measured in body axis coordinates, and has components l_x, l_y and l_z. Define ω_B as the angular body rate vector of the IRS, as expressed in body axis coordinates. It is directly available from the LTN-90-100 dataport and has components ω_{Bx}, ω_{By}, ω_{Bz}. Corrected Doppler velocity components (i.e. U_{DC}, V_{DC}, W_{DC}) are then calculated as (see Appendix C for details)

$$\begin{aligned} U_{DC} &= U_D - \omega_{Bz}\,l_y + \omega_{By}\,l_z \\ V_{DC} &= V_D + \omega_{Bz}\,l_x - \omega_{Bx}\,l_z \\ W_{DC} &= W_D - \omega_{By}\,l_x + \omega_{Bx}\,l_y \end{aligned} \qquad (8)$$

The three Doppler-based measurement components available at discrete times t_k, $z_3(k)$, $z_4(k)$ and $z_5(k)$, can then be written as

$$\begin{aligned} z_3(k) &= U_I(k) - U_{DC}(k) \\ z_4(k) &= V_I(k) - V_{DC}(k) \\ z_5(k) &= W_I(k) - W_{DC}(k) \end{aligned} \qquad (9)$$

The 24-element rows of **H** that are associated with these three measurements (i.e. rows 3, 4 and 5) have the following forms:

$$\mathbf{H}_3 = (0\ 0\ 0\ h_{3,4}\ h_{3,5}\ h_{3,6}\ 0\ldots 0\ h_{3,20}\ 0\ 0\ h_{3,23}\ h_{3,24})$$

$$\mathbf{H}_4 = (0\ 0\ 0\ h_{4,4}\ h_{4,5}\ h_{4,6}\ 0\ldots 0\ h_{4,21}\ 0\ h_{4,23}\ h_{4,24})$$

$$\mathbf{H}_5 = (0\ 0\ 0\ h_{5,4}\ h_{5,5}\ h_{5,6}\ 0\ldots\ldots 0\ h_{5,22}\ 0\ 0) \qquad (10)$$

where the indicated non-zero elements are defined in Appendix C.

The noise variances assumed for these three measurements, r_3, r_4 and r_5, comprise diagonal elements 3, 4 and 5 of the 5 x 5 measurement noise covariance matrix **R**. Nominal values chosen for them are $r_3 = (5.34\ \text{ft/s})^2$, $r_4 = (10.68\ \text{ft/s})^2$ and $r_5 = (5.34\ \text{ft/s})^2$, based on an analysis of typical Doppler radar data from the Twin Otter. Doppler-based measurements are available at a 2 Hz rate, much faster than the basic Kalman filter update rate of 0.1 Hz.

Due to the basic nature of Bierman's **UDU**^{T} factorization algorithms, the individual Loran and Doppler-based measurement components and their associated statistics (i.e. z_i and $\mathbf{H}_i$, r_i) are processed in a one-at-a-time fashion. Recall that the software algorithms for doing this form of Kalman filter processing have been described in subsection 2.1.

3.4 Measurement Averaging (Prefiltering)

Kalman filter measurement averaging (or prefiltering, as it is sometimes called - see Ref. 8) is a technique used to take full advantage of discrete measurement data, even when it is available at a much higher sample rate than the basic update rate of the Kalman filter itself. Rather than discarding all of the measurement information 'in between' Kalman filter time update points, every measurement data point is used. The mathematical details of this concept are given in Appendix D. However, a short description of the measurement averaging concept is given below.

The use of the measurement averaging concept affects the way in which **z**, **H**, **v** and **R** are computed, as well as the order in which the five Kalman filter recursion equations (i.e. Eqns. 3) are invoked. Let $\Delta T = t_{k+1} - t_k$ be the (constant) Kalman filter update interval, and assume that the vector measurements occur at N equally spaced time points within ΔT (i.e. $z(t_k + \delta t_i)$ where $\delta t_i = i\,\Delta T/N$; i = 1,2,..,N). Then define the following <u>averaged</u> quantities:

$$\mathbf{z}_{av}(k) = (1/N) \sum_{i=1}^{N} \mathbf{z}(t_k + \delta t_i)$$

$$\mathbf{H}_{av}(k) = (1/N) \sum_{i=1}^{N} [\mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)]$$

$$\mathbf{v}_{av}(k) = (1/N) \sum_{i=1}^{N} \mathbf{v}(t_k + \delta t_i)$$

$$\mathbf{R}_{av}(k) = (1/N^2) \sum_{i=1}^{N} \mathbf{R}(t_k + \delta t_i) = (1/N)\,\mathbf{R} \qquad (11)$$

where $\mathbf{z}_{av}(k)$ is the averaged measurement, $\mathbf{H}_{av}(k)$ is the averaged observation matrix, $\Phi(t_k, t_k + \delta t_i)$ is the so-called intermediate transition matrix, $\mathbf{v}_{av}(k)$ is the equivalent averaged measurement noise and $\mathbf{R}_{av}(k)$ is the averaged measurement noise covariance matrix. All of these averaged quantities are referenced in time to t_k, and $\mathbf{R}_{av}(k)$ can be simplified as shown above when it is assumed to be a constant matrix. With the foregoing definitions, the equivalent averaged observation matrix is

$$\mathbf{z}_{av}(k) = \mathbf{H}_{av}(k)\ \delta\mathbf{x}(k) + \mathbf{v}_{av}(k) \qquad (12)$$

Kalman Filter Sequence For Prefiltering -

The Kalman filter equation sequence, as shown in Eqns. 3, must be modified when prefiltering takes place. The filter sequence with prefiltering proceeds as follows:

i) At current time t_{k+1}^- (i.e. the time just prior to processing the latest averaged measurements) the Kalman filter acquires an averaged measurement vector $\mathbf{z}_{av}(k)$ from the prefiltering process, with $\mathbf{z}_{av}(k)$ computed by averaging the measurements sampled over the update interval $t_k \dashrightarrow t_{k+1}$.

ii) The Kalman filter processes each element of $\mathbf{z}_{av}$ sequentially (i.e. as a scalar measurement), using the **UDUPDC** subroutine, in order to compute measurement updates, $\delta\mathbf{x}$ and $\mathbf{P}$, referenced to time t_k^+.

iii) The Kalman filter estimates, $\delta\mathbf{x}(k^+)$ and $\mathbf{P}(k^+)$, are then time-extrapolated to the current time t_{k+1} using subroutine **MWGSUD**. The time update of $\delta\mathbf{x}$, as computed by **MWGSUD**, is equivalent to the equation

$$\delta\mathbf{x}(k+1^-) = \Phi(k,k+1)\ \delta\mathbf{x}(k^+) \qquad (13)$$

Estimates of the error state vector $\delta\mathbf{x}(k+1^-)$ are then in time synchronization with the latest values of the navigation variables available from any of the navaids.

iv) In the case of the assumed feedforward Kalman filter error state configuration, $\delta\mathbf{x}(k+1^-)$ contains the cumulative error build-up in the IRS parameters of interest at t_{k+1}. These IRS parameters can be corrected at the basic filter rate of 0.1 Hz, or even at the higher IRS rate of 16 Hz (i.e. by applying the last best estimates of $\delta\mathbf{x}$ over the entire 10 second filter update interval). If the 10 second 'resets' in the corrected 16 Hz IRS digital parameters appear too prominently, there is even the possibility of low-pass filtering (i.e. digitally) a 16 Hz version of the correction signal in order to smooth out the high frequency component of the resets.

3.5 Euler Angle Error Correction

The Kalman filter error state estimates for IRS position and velocity errors are computed in an appropriate geographic coordinate frame for direct correction of the corresponding IRS parameters. However, in the case of the IRS attitude errors (i.e. Euler angle errors) a corrector algorithm must be used

in order to transform Kalman filter estimates of tilt errors into equivalent estimates of Euler angle errors. The fundamental tilt errors being estimated by the Kalman filter, $\boldsymbol{\varepsilon} = [\varepsilon_N \ \varepsilon_E \ \varepsilon_z]^T$, are the attitude errors of the so-called computed geographic coordinate frame (i.e. local level, N-E-z) relative to the true geographic frame. A corrector algorithm converts tilt errors into equivalent errors in the direction cosine matrix (DCM) that describes the transformation from body axes to geographic axes, denoted by $\mathbf{C}_b{}^G$. Once the DCM has been corrected, it is quite straightforward to compute corrected values of the associated Euler angles (i.e. θ, ϕ, ψ).

Consider $\boldsymbol{\varepsilon}$ to be a small angle error vector representing the small rotation of the computed geographic frame about the true geographic (N, E, z) axes. Define $\boldsymbol{\varepsilon x}$ as the matrix-equivalent of the so-called skew symmetric cross product operator (Ref. 13) that can be associated with $\boldsymbol{\varepsilon}$. Then $\boldsymbol{\varepsilon x}$ has the following form:

$$\boldsymbol{\varepsilon x} = \begin{bmatrix} 0 & -\varepsilon_z & \varepsilon_E \\ \varepsilon_z & 0 & -\varepsilon_N \\ -\varepsilon_E & \varepsilon_N & 0 \end{bmatrix} \tag{14}$$

Let $\mathbf{C}_b{}^{G^*}$ be the error-corrupted version of the DCM which results from errors in the Euler angles being measured by the IRS. In Appendix A it is shown how $\mathbf{C}_b{}^{G^*}$ is computed using the IRS-supplied Euler angles θ, ϕ and ψ. Let $\mathbf{C}_{bC}{}^G$ be a corrected version of this DCM, based on knowledge of $\boldsymbol{\varepsilon}$. Then the error in $\mathbf{C}_b{}^{G^*}$, $\delta\mathbf{C}_b{}^G$, would be defined as

$$\delta\mathbf{C}_b{}^G = \mathbf{C}_b{}^{G^*} - \mathbf{C}_b{}^G \tag{15}$$

where $\mathbf{C}_b{}^G$ is the true DCM. It can be shown (Ref. 13) that, to first order,

$$\delta\mathbf{C}_b{}^G = -(\boldsymbol{\varepsilon x})\ \mathbf{C}_b{}^G \tag{16}$$

Then a corrected version of $\mathbf{C}_b{}^{G^*}$, $\mathbf{C}_{bC}{}^G$, can be calculated as

$$\begin{aligned} \mathbf{C}_{bC}{}^G &= \mathbf{C}_b{}^{G^*} - \delta\mathbf{C}_b{}^G \\ &= \mathbf{C}_b{}^{G^*} + (\boldsymbol{\varepsilon x})\,\mathbf{C}_b{}^G \\ &\equiv \mathbf{C}_b{}^{G^*} + (\boldsymbol{\varepsilon x})\,\mathbf{C}_b{}^{G^*} \\ &\equiv (\mathbf{I} + \boldsymbol{\varepsilon x})\,\mathbf{C}_b{}^{G^*} \end{aligned} \tag{17}$$

to first order.

Once $\mathbf{C_b}^{G^*}$ has been corrected, as above, corrected values for the Euler angles can be computed as

$$\theta_C = \arctan [\, C_{3,1C}/(1 - C_{3,1C}^2)^{1/2} \,]$$

$$\phi_C = \arctan [\, C_{3,2C}/C_{3,3C} \,]$$

$$\psi_C = \arctan [\, C_{2,1C}/C_{1,1C} \,] \qquad (18)$$

From Eqns. 18, it should be noted that only five of the nine DCM elements in $\mathbf{C_b}^G$ need to be corrected in order to correct the Euler angles, namely $C_{1,1}$, $C_{2,1}$, $C_{3,1}$, $C_{3,2}$ and $C_{3,3}$.

Computation of Euler Angle Error Bounds -

As well as the calculation of θ_C, ϕ_C, ψ_C, using Eqns. 18 above, it would be useful to have an equivalent 1-sigma uncertainty bound for each of the Euler angle calculations, just as we do for the other Kalman filter error state estimates. This is not straightforward, because errors in the Euler angles do <u>not</u> have a direct, one-to-one correspondence with the tilt error states (for which the Kalman filter does compute an updated covariance). However, an ad hoc procedure that should give a representative uncertainty bound for each of the Euler angles would be as follows:

1) For each of the three tilt error bounds, set the corresponding tilt error component to that value (with the other two components set to zero) and calculate a value of $\mathbf{C_{bC}}^G$ using Eqn. 17. Repeat the procedure using the negative of the bound in each case. The result will be six different values of $\mathbf{C_{bC}}^G$.

2) For each of the six $\mathbf{C_{bC}}^G$'s, compute 'corrected' values for the three Euler angles using Eqn. 18. There will be six values for each of the three Euler angles.

3) For each 'corrected' value, compute an Euler angle error by differencing the 'corrected' value with the corresponding IRS-based Euler angle. The result will be six different error values for each of the three Euler angles.

4) For each of the three Euler angles, do an RSS (i.e. square root of the sum of the squares) of the six values in order to arrive at a representative value for the bound.

Computation of True Tilt Errors -

When using <u>simulated</u> flight test data (thus having access to the true aircraft trajectory and true navigation system errors being simulated), it would be beneficial to be able to derive the true IRS tilt errors from knowledge of the Euler angle errors. Equations 15 and 16, when combined, yield

$$(\varepsilon x)\, \mathbf{C_b}^G = \mathbf{C_b}^G - \mathbf{C_b}^{G^*}$$

or

$$(\varepsilon x)\, \mathbf{C_b}^G (\mathbf{C_b}^G)^{-1} = \mathbf{I} - \mathbf{C_b}^{G^*} (\mathbf{C_b}^G)^{-1}$$

or

$$\varepsilon x = I - C_b{}^{G'} (C_b{}^{G})^T, \text{ since } (C_b{}^{G})^{-1} = (C_b{}^{G})^T \quad (19)$$

Note that $C_b{}^{G'}$ is the DCM that would be calculated from error-corrupted simulated IRS attitude data. On the other hand, $C_b{}^{G}$ would be calculated using the reference trajectory attitude (i.e. the true attitude). Once the skew symmetric matrix (εx) is calculated from Eqn.19, the matrix elements that correspond to the tilt errors ε_N, ε_E and ε_z can be identified and compared with Kalman filter estimates of the same tilt error parameters.

4.0 SOFTWARE FOR NAVIGATION SIMULATION AND KALMAN FILTERING

In order to analyze and refine proposed Kalman filter integrated navigation designs properly, accurate simulation studies must first be conducted. The heart of these simulation studies is a set of algorithms for generating 'real world' data for the various navigation sensors, based upon a specified aircraft flight profile and known error statistics for each sensor. A comprehensive set of navigation sensor simulation software has been developed at the Flight Research Laboratory to model, quite accurately, realistic aircraft flight trajectory dynamics, and the error dynamics of such airborne navigation sensors as medium accuracy INS's, Doppler velocity sensing (DVS) systems and various radio position fixing/ranging systems. This section describes a set of simulation software modules that have been written for execution under the IBM VM Operating System resident on NRC's IBM 3090 mainframe computer. Results will be given to show, for a typical simulated flight trajectory, how realistic navigation sensor data are created. A general description will be given of the comprehensive Kalman filtering software package that has been developed to run on either simulated or real navigation data from the Twin Otter's suite of navigation sensors. A series of plots derived from the Kalman filter running on simulated data will be shown to document the performance and accuracy of the filter design. Finally, the section will conclude with a list of the general findings from the extensive series of simulation studies that were conducted.

4.1 Software for Generating Reference Trajectory Profile Data

a) PROFST

A VM Executive routine, **PROFST EXEC**, is used to run a trajectory profile generation FORTRAN program with the FORTRAN name **PROFST**. **PROFST** creates 3-dimensional inertial flight trajectory data (in metric units) at specified points in time using a user-supplied description of the desired profile. The resulting trajectory profile dataset is then used as the reference for the true aircraft trajectory when analyzing Kalman filter accuracy and it is also the starting point for creating realistic, error-corrupted navigation data from the various navaids being simulated. This program calls a series of general-purpose profile generation subroutines (i.e. **PROGEN**, **PROGNI**, **SEVAL**, **SPLINE**), which are based on the use of cubic splines. **PROFST** takes the inertial velocity data supplied by the **PROGEN** subroutine and integrates it, using Simpson's rule of integration, to generate 3-dimensional flight trajectory position data. **PROFST** also creates appropriate output files containing all of the flight trajectory parameters of interest. When running the program, the user specifies the desired sample rate (in Hz) of the trajectory data to be created and the total time length (in seconds) of the trajectory. The input and output files involved in the execution of **PROFST** have the following characteristics:

Input File: A dataset with the generic name **PROGD DATAn** (where n is a dataset number to denote the particular type of trajectory file being created) is required. **PROGD DATAn** specifies the starting 3-dimensional position (Lat, Long, altitude) for the trajectory in its first record, and then consists of records of desired aircraft dynamic state (i.e. attitude and speed) at selected points along the trajectory, to a maximum of 100 trajectory points. During portions of the trajectory where the aircraft state changes rapidly the trajectory points are more closely spaced than during 'benign' sections of the trajectory. This approach ensures that a realistic trajectory is 'captured' by the cubic spline fitting process during the high dynamics portions of the simulated flight trajectory.

Output Files: **PROFST** takes the sparse set of trajectory points defined in the input dataset **PROGD DATAn** and interpolates between the points using cubic spline techniques. Two output files are normally created from a **PROFST** run, as follows:

SPUNFMT OUT33 - contains 20 flight parameters at a 64 Hz data rate (unformatted, single precision), with 64 Hz chosen as the minimum rate possible in order to generate strapdown IMU data with sufficient accuracy. After the idealized strapdown IMU data is generated, this rather large file is no longer needed and is erased to save disk storage space.

SPUNFMT TWOHZ - contains the same 20 flight parameters as **SPUNFMT OUT33** (unformatted, single precision), but at the lower rate of 2 Hz. This file is a lot smaller than **SPUNFMT OUT33** and is retained for use as the reference trajectory data set for subsequent Kalman filter analysis purposes.

Both output files, **SPUNFMT OUT33** and **SPUNFMT TWOHZ**, contain the same set of 20 aircraft inertial flight trajectory parameters, but at the differing data rates. For each dataset the 20 parameters are written in the following order:

- Trajectory point number (i.e. record counter)
- Time from start of flight
- 3 components of aircraft velocity (north, east, down in m/s)
- 3 components of aircraft velocity rate (north, east, down in m/s^2)
- 3 components of aircraft attitude (roll, pitch, heading in rad)
- 3 components of aircraft attitude rate (roll rate, pitch rate, yaw rate in rad/s)
- 3 components of position change from start location (ΔLat, ΔLong, Δalt in deg, deg, m)
- 3 components of absolute aircraft position (Lat, Long, alt in deg, deg, m)

b) PROGPLT

PROGPLT EXEC is a menu-driven Executive routine to plot various subsets of trajectory profile parameters that have been created by a **PROFST** run that produces a **SPUNFMT OUT33** data set. **PROGPLT** calls a general-purpose, FORTRAN-based time series plotting program, **PLOTTER** (with associated subroutines **PLOTT** and **BGPLT**), which has been written at the Flight Research Laboratory and utilizes the **DISSPLA** package of plotting subroutines. There are five different subsets of flight trajectory parameters that can be plotted, namely i) the 3 inertial velocity components (note that metric units are used), ii) the 3 aircraft velocity rate components, iii) the 3 aircraft attitude components (i.e. Euler angles), iv) the 3 attitude rate components and v) the 3 change-in-position components. Figures 4 to 8 show a series of plots generated by **PROGPLT** for a 1 hour simulated flight profile corresponding to a square pattern with take-off and landing included as well.

4.2 Software for Generating Realistic Navigation Data

A series of programs has been written to take the ideal flight trajectory data, model realistic levels of time-varying errors in each of the navaids to be simulated, and generate data files of simulated navigation data in formats similar to real navigation data that would be acquired onboard the aircraft. At the moment, the software is configured to handle generic versions of a strapdown INS, Doppler velocity sensor and radio position fixing sensor. For this report, the software has been used to accurately simulate an LTN-90-100 IRS, a Decca Type 72 Doppler radar and an ARNAV R-40 Loran-C receiver.

a) SDIMU

By far the most complex navigation simulation software to develop is that required to accurately simulate an IMU and/or INS. **SDIMU EXEC** is an Executive routine that is used to run **SDIMU**, a FORTRAN program that simulates ideal (i.e. perfectly accurate) data being generated from a strapdown IMU, as well as ideal height (i.e. altimetry) information. The raw outputs from a strapdown IMU are the body frame velocity and attitude increments (i.e. changes in velocity and attitude over a very small time interval) usually sampled at a fairly high data rate. For simulation purposes, IMU increments are calculated at a 64 Hz rate by running **SDIMU** using the inertial rate information available from the **SPUNFMT OUT33** reference trajectory data file as input. A small input control file, **SDIMU CTRL**, specifies the strapdown IMU sample rate and the time length desired for the run. The standard output from **SDIMU** is another 64 Hz single precision, unformatted data file, **SDIMU DATA**, which consists of 8 parameters, namely i) a record counter, ii) 3 components of incremental change in body axis velocity (m/s), iii) 3 components of incremental change in body axis attitude (rad) and iv) true altitude (m). As previously stated, upon completion of a **SDIMU** run, the 64 Hz input file, **SPUNFMT OUT33**, which is not required for subsequent analysis purposes, is erased to save disk storage space.

b) SDNAV1

An Executive routine, **SDNAVG1 EXEC**, controls the execution of **SDNAV1**, a FORTRAN program that processes the strapdown IMU incremental angles and velocities in order to emulate a strapdown INS or IRS (such as the LTN-90-100) computing inertial position, velocity and attitude. In **SDNAV1**, the navigation equations are solved using a wander azimuth navigator mechanization, with all updates occurring at a 64 Hz rate (as is done in the LTN-90-100). However, the standard output rate for the simulated INS data is 16 Hz which is the rate at which LTN-90-100 data are sampled onboard the Twin Otter. Included in **SDNAV1** is code for modelling realistic errors in a baro altimeter source as well as an emulation of the third-order baro damping loop that is implemented inside the LTN-90-100. The **SDNAV1** software was developed at Flight Research as a major re-structuring of a DREO software package called **NAVGTR** and utilizes DREO-developed subroutines **QCINI**, **CORIOL**, **QCUPDT** and **NAVCAL** for performing the various standard strapdown navigation calculations. There are 2 input files required in order to run **SDNAV1** - a control file, **SDNAV1 CTRL**, and the data file, **SDIMU DATA**, of ideal strapdown IMU increments sampled at 64 Hz. **SDNAV1 CTRL** contains the following information: i) input/output sample rates, ii) time length of the run, iii) initial conditions on aircraft position, velocity and attitude and iv) statistical information on gyro, accelerometer and altimeter errors. Initial conditions can be specified in this input file to simulate typical INS start-up errors arising from the alignment process. The statistical information required to model gyro and accelerometer errors properly is the kind shown in Table 3.2 of Section 3 for the gyros and accelerometers in the LTN-90-100. A combination of bias, ZMWG noise and first-order Markov error modelling is used in **SDNAV1** to produce realistic time varying errors in the gyros, accelerometers and altimeter. The standard output file created by running **SDNAV1** is **TWOTTER INSDATA**, a data file consisting of 16 INS parameters sampled at 16 Hz in the same order,

and with the same engineering units, as real LTN-90-100 IRS data that would be collected onboard the Twin Otter (i.e. as shown in Table 3.1 of Section 3).

c) SDERKF

A FORTRAN program, **SDERKF**, is used to compute (for plotting purposes) the differences between the simulated strapdown INS parameters, as created by **SDNAV1** in its output data set **TWOTTER INSDATA**, and the original inertial reference parameters created by **PROFST** in its output file **SPUNFMT TWOHZ**. The execution of **SDERKF** is normally controlled by the same Executive routine that is used to control **SDNAV1**, namely **SDNAVG1 EXEC** - i.e. a run of **SDNAV1** is immediately followed by a run of **SDERKF** via **SDNAVG1 EXEC**. **SDERKF** requires 3 input files in order to run: **SDNAV1 CTRL**, **TWOTTER INSDATA** and **SPUNFMT TWOHZ**. It produces one main output file, **SPUNFMT ERR**, which consists of a record counter plus the 9 true inertial error quantities corresponding to the simulated IRS system (3 position component errors, 3 velocity component errors and 3 attitude component errors) all sampled at a 0.1 Hz rate. The engineering units chosen for the inertial errors are the same ones being used for Kalman filter output display purposes (see subsection 4.4).

d) ERRPLOT

ERRPLOT EXEC is a menu-driven Executive routine for controlling the plotting of various subsets of simulated IRS errors using the Flight Research general-purpose time series plotting package, **PLOTTER**. The user can chose any of the following three subsets of true IRS errors to be plotted: i) the 3 IRS position error components, ii) the 3 IRS velocity error components or iii) the 3 IRS attitude error components. Figures 9 to 11 show typical plotted outputs for each of these subsets. In this particular case, the flight trajectory consists of a 1 hour square pattern, including takeoff and landing phases. **ERRPLOT** has been used to confirm the accuracy of the IRS simulation software by comparing plotted outputs with those for known IRS behaviour (Refs. 14, 15) under the same set of initial conditions and sensor errors.

e) DOPLOR

DOPLOR EXEC is used to control the execution of **DOPLOR**, a FORTRAN-based program that creates realistic simulated navigation data, based on a desired reference trajectory, that corresponds to data from a Decca Type 72 Doppler radar and an ARNAV R-40 airborne Loran-C receiver, including data rates and engineering units identical to those for real data from the Twin Otter data acquisition system. **DOPLOR** requires 2 input files in order to run - a control file, **DOPLOR CTRL**, and the 2 Hz reference trajectory data file, **SPUNFMT TWOHZ**, that has been created by **PROFST**. **DOPLOR CTRL** contains the following information required for the proper simulation of Doppler and Loran-C data: i) time length of the run and time interval for overwater flight, ii) Doppler-to-INS lever arms, iii) Loran-C latitude and longitude error statistics and iv) Doppler error statistics. Loran-C error statistics consist of ZMWG noise and first-order Markov specifications, while Doppler error statistics consist of first-order Markov processes for each of U channel scale factor, V and W channel boresights and ZMWG noise in each of the U, V and W channels. As well, if overwater flight is involved, then first-order Markov processes are defined for the N and E sea bias components. Typical values for these various statistics can be found in Tables 3.5 and 3.6 and sub-subsection 3.3.3 of Section 3. Two output files are created from a **DOPLOR** run - i) **TWOTTER LORDATA** which contains realistic, error-corrupted Loran-C Lat and Long data and the true Loran-C

Markov error processes, all at the standard Loran-C sample rate of 1 Hz and ii) **TWOTTER DOPDATA**, containing realistic, error-corrupted Doppler U, V, W channel data and the true Doppler Markov error processes, all at the standard Decca Doppler sample rate of 2 Hz. The true Markov error processes for the simulated Loran-C and Doppler data become useful for assessing the accuracy of a Kalman filter algorithm when it is trying to estimate these particular Markov error states.

4.3 Kalman Filter Software Descriptions

A comprehensive set of Kalman filter software has been developed at Flight Research, based on the Kalman filter design described in Section 3, for use with Twin Otter navigation data in either real or simulated form. The purpose of this subsection is to outline the algorithmic and hierarchical structure of the Kalman filter software down to the level of the individual subroutines that are called and the input/output datasets that are normally required.

a) SDKFNAV

SDKFNAV EXEC is a menu-driven Executive routine for controlling the execution of four slightly different versions of the Kalman filter software. These four different versions, called by **SDKFNAV EXEC** under user control, have the following main FORTRAN program names: i) **SDKFN1** - for running the Kalman filter on real data with the measurement averaging option (see subsection 3.4 for details), ii) **SDKFN2** - for running the Kalman filter on real data, but with no measurement averaging, iii) **SDKFN3** - for running the Kalman filter on simulated data with the measurement averaging option and iv) **SDKFN4** - for running the Kalman filter on simulated data with no measurement averaging. Within **SDKFNAV EXEC**, one of two other EXEC's is invoked depending on whether real or simulated data is to be used - i) **SDKFFDEF EXEC** defines all of the input and output files that are required for the simulated data case and ii) **SDKFRFDF EXEC** defines all of the input and output files for the real data case. The following are very brief descriptions of the various input and output files that are required for the various versions of the Kalman filter as well as descriptions of the complete set of FORTRAN modules that comprise the Kalman filter software package:

Input Files:

SDKFNAV CTRL - passes, to any version of **SDKFNn**, all of the control information that is required to initialize the filter, including: i) time length of the filter run and time interval for overwater flight, ii) nominal starting latitude and Doppler-to-INS lever arms, iii) IRS system error state statistics (i.e. Table 3.4), iv) sensor Markov error statistics (i.e. Table 3.5), v) IRS system plant noise statistics (i.e. Table 3.6) and vi) measurement noise statistics (given in sub-subsection 3.3.3). As well, for the real data case there may be further information supplied at the end of this file concerning Visual On-Top (VOT) position reference points along the flight path.

TWOTTER LORDATA - the 1 Hz version of airborne Loran-C data, having the identical format for both the real and simulated data cases, except that in the simulated data case the true Loran-C Markov error processes are added on to the end of each record for Kalman filter analysis purposes.

TWOTTER DOPDATA - the 2 Hz version of the Decca Doppler radar data, having identical formats for both the real and simulated data cases, except that in the simulated data case the true Doppler Markov error processes are added on to the end of each record for Kalman filter analysis purposes.

TWOTTER INSDATA - the 16 Hz version of the Litton LTN-90-100 IRS data, as defined in Table 3.1, having identical formats for both the real and simulated data cases.

SPUNFMT ERR - the 0.1 Hz version of true IRS system errors, as produced by program **SDERKF**, and used by the Kalman filter program, **SDKFNn** (n = 3 or 4), to evaluate its state estimation accuracy. This input file is only used for the simulated data case.

Output Files:

TWOTTER MEASURE - a 0.1 Hz data file consisting of the five scalar measurements being processed by the Kalman filter, namely i) 2 measurements based on Loran-C and ii) 3 measurements based on Doppler radar (see sub-subsection 3.3.3 for details). It is instructive to plot this measurement data to confirm that it looks reasonable for processing by the Kalman filter.

TWOTTER SYSER1 - a 0.1 Hz unformatted data file of filter-estimated IRS system errors and associated +/- 1-sigma bounds, for plotting purposes. This particular file contains information on the first 6 IRS system error states, as shown in Table 3.4.

TWOTTER SYSER2 - a 0.1 Hz unformatted data file, similar to **TWOTTER SYSER1**, but containing information on the last 4 IRS system error states shown in Table 3.4, for plotting purposes.

TWOTTER MARKER1 - a 0.1 Hz unformatted data file of filter-estimated sensor Markov error states and associated +/- 1-sigma bounds, for plotting purposes. This particular file contains information on the first 7 Markov error states, as shown in Table 3.5.

TWOTTER MARKER2 - a 0.1 Hz unformatted data file, similar to **TWOTTER MARKER1**, containing information on the last 7 sensor Markov error states shown in Table 3.5 for plotting purposes.

TWOTTER INERRS - a small file containing IRS horizontal position error information at each of the Visual On-Top (VOT) points overflown during an actual Twin Otter navigation flight. This output file is only created for the real data case and then used to plot IRS position errors.

TWOTTER LRERRS - a small file containing Loran-C horizontal position error information at each of the VOT points overflown during an actual navigation flight. This file is only created for the real data case and used for plotting actual Loran-C position errors during a flight.

TWOTTER KFERRS - a small file containing filter-corrected IRS horizontal position error information at each of the VOT points overflown during an actual flight. This file is only created for the real data case and used for determining the true position

accuracy of the Kalman filter.

TWOTTER KFBNDS - a 0.1 Hz data file of filter-generated +/- 1-sigma bounds for the IRS horizontal position error states. This information, only created in the real data case, is plotted along with that of **TWOTTER KFERRS** in order to evaluate Kalman filter performance.

FORTRAN Module Descriptions:

SDKFNn - this is the main Kalman filter program, from which the various subroutines associated with specific filtering tasks are called. As already stated, there are actually 4 slightly different versions of the Kalman filter software (in order to handle the options of real or simulated data and prefiltering or no prefiltering), with most of the program coding differences taking place in the 4 versions of main program **SDKFNn** (n = 1,. . . ,4). The main program sets up the specified initial conditions for the filter run, controls the 0.1 Hz rate at which measurements are defined and the various Kalman filter updates take place and writes out the pertinent data to the various output files that are created.

SDPHGM - a subroutine, called by **SDKFNn** at a 2 Hz rate, that calculates the time varying portion of the so-called intermediate Φ transition matrix (see Appendix D) and intermediate **G** matrix using trapezoidal integration on the 16 Hz inertial data. A slightly different version of this subroutine, **SDPHG1**, is used when measurement averaging (i.e. prefiltering) is involved.

SDFGCL - a subroutine, called by **SDPHGM**, that computes the individual matrix elements of **F** and **G** according to the equations given in Appendix B. As well, the latest elements of the DCM (see Appendix A) are computed. In the case of prefiltering, a slightly modified version of this subroutine, **SDFGC1**, is used.

SDDRCL - a subroutine, called by **SDKFNn** at a 2 Hz rate, that calculates the IRS/Doppler velocity difference measurements, including lever arm corrections (see Appendix C for details). A slightly modified version of this subroutine, **SDDCLC**, is used for the simulated data case.

SDHCAL - a subroutine, called by **SDKFNn** at a 0.1 Hz rate, that calculates the non-zero elements of the $\mathbf{H}_i$ vectors (i = 3, 4, 5) required for Doppler-based measurement processing. A slightly different version of this subroutine, **SDHCLC**, is used for the prefilter case.

FACTOR - a general-purpose Kalman filter subroutine, called by **SDKFNn** and described in subsection 2.2, for the efficient factorization of the error state covariance matrix **P** into Bierman's $\mathbf{UDU}^T$ form.

UNFACT - a general-purpose Kalman filter subroutine, called by **SDKFNn** and described in subsection 2.2, for reconstructing a **P** matrix from its $\mathbf{UDU}^T$ decomposition form. It is used to derive the +/-1-sigma bounds of the various error state estimates, at each Kalman filter update, for display purposes.

MWGSUD - a general-purpose Kalman filter subroutine, called by **SDKFNn** and described in subsection 2.2, for performing Bierman's efficient time update of the error state estimate **x** and its factored covariance matrix **U**.

UDUPDC - a general-purpose Kalman filter subroutine, called by **SDKFNn** and described in subsection 2.2, for performing Bierman's efficient measurement update of **x** and **U** based on individual scalar measurements.

NMDLAT - a small subroutine used in **SDKFNn** for accurately converting a change in latitude, as expressed in units of nm, into the equivalent change in units of geographical degrees by using local earth's curvature information.

NMDLNG - a small subroutine used in **SDKFNn** for accurately converting a change in longitude, as expressed in units of nm, into the equivalent change in units of geographical degrees using local earth's curvature information.

CONLAT - a small subroutine used in **SDKFNn** for accurately converting a change in latitude, as expressed in units of geographical degrees, into the equivalent change in units of nm using local earth's curvature information.

CONLNG - a small subroutine used in **SDKFNn** for accurately converting a change in longitude, as expressed in units of geographical degrees, into the equivalent change in units of nm using local earth's curvature information.

b) MESPLT

MESPLT EXEC is a menu-driven Executive routine for controlling the execution of the FORTRAN plotting program, **MESPLT**, that plots the 2 different types of Kalman filter measurement data being processed, namely i) the Loran-C/IRS position difference measurements (z_1 and z_2 in sub-subsection 3.3.3) and ii) the Doppler/IRS velocity difference measurements (z_3, z_4 and z_5 in sub-subsection 3.3.3). The data to be plotted is contained in file **TWOTTER MEASURE** which is created by **SDKFNn**.

c) VOTPLT

VOTPLT EXEC is a menu-driven Executive routine for controlling the execution of the FORTRAN program, **VOTPLT**, that plots any or all of the following sets of errors: i) Kalman filter horizontal position errors at the VOT points, ii) IRS horizontal position errors at the VOT points and iii) Loran-C horizontal position errors at the VOT points. **VOTPLT** is only used for the real data case and requires the following input data files as created by **SDKFNn**: **TWOTTER KFERRS**, **TWOTTER KFBNDS**, **TWOTTER INERRS** and **TWOTTER LRERRS**.

d) SDKFPLT

SDKFPLT EXEC is a menu-driven Executive routine for controlling the plotting of various subsets of Kalman filter-generated error state estimates and associated +/- 1-sigma bounds using the Flight Research general-purpose time series plot program, **PLOTTER**. The following subsets of parameters can be chosen by the user for plotting:

I) filter estimates of the 3 IRS position error components (or actual filter errors in measuring the IRS position error states, for the simulated data case) plus associated +/- 1-sigma bounds,

ii) filter estimates of the 3 IRS velocity error components (or actual filter errors in measuring the IRS velocity error states, for the simulated data case) plus associated +/- 1-sigma bounds,

iii) filter estimates of the 3 IRS tilt error components (or actual filter errors in measuring the tilt errors, for the simulated data case) and associated +/- 1-sigma bounds,

iv) filter estimates of the 3 IRS gyro Markov error components and associated +/- 1-sigma bounds,

v) filter estimates of the 3 IRS accelerometer Markov error components and the associated +/- 1-sigma bounds,

vi) filter estimates of the 2 Loran-C Markov error components and the 2 sea bias Markov error components (or actual filter errors in estimating these components, for the simulated data case) plus associated +/- 1-sigma bounds,

vii) filter estimates of the 3 Doppler Markov error components (or actual filter errors in estimating these components, for the simulated data case) and associated +/- 1-sigma bounds,

viii) actual Loran-C Markov error processes (for the simulated data case only)

ix) actual Doppler Markov error processes (for the simulated data case only).

The following datasets are required in order to invoke all options of **SDKFPLT**: **TWOTTER SYSER1**, **TWOTTER SYSER2**, **TWOTTER MARKER1**, **TWOTTER MARKER2**, **TWOTTER LORDATA**, **TWOTTER DOPDATA**.

e) INNPLT

INNPLT EXEC is a menu-driven Executive routine for controlling the execution of the FORTRAN plotting program, **INNPLT**, that plots the 5 measurement residual time series (described in subsection 4.4).

4.4 Kalman Filter Results Using Simulated Data

Not a lot of emphasis will be placed on these simulation results, since the most important results are certainly those demonstrating Kalman filter performance for real navigation data taken under typical operating conditions. However, in order to demonstrate that the fundamental Kalman filter design is sound, and performs according to the theory, some simulated data results will be shown. This will also serve as a demonstration of the full capabilities of the navigation simulation/Kalman filtering software package.

Kalman filter results will be shown for the same one hour simulated flight scenario that was used to show example output plots from the **ERRPLOT** plotting routine (i.e. a flight trajectory consisting of a square pattern together with takeoff and landing phases - see Figs. 4 to 8). Table 4.1 gives the details of the IRS initialization errors and sensor statistical errors that were used in control file **SDNAV1 CTRL** when running **SDNAV1** to produce the simulated IRS navigation data having the time varying errors shown in Figs. 9 to 11. From these figures it can be seen that there is a significant build-up in IRS latitude and north velocity error in particular; however, these and the other IRS error levels are still realistic (both in their initial values and variations with time), based on the specifications given by Litton for the LTN-90-100 IRS.

Table 4.2 shows the statistical error specifications used for simulating Loran-C and Doppler navigation data for the same flight trajectory as defined above for simulated IRS data. Figures 12 to 13 show the simulated first-order Markov error time histories that define the time varying errors in each of these navaids. The plots shown in these figures were created using appropriate options within the **SDKFPLT** plotting package described in subsection 4.3. The fundamental time varying nature of these Markov simulations has been compared with the known error behaviour of the actual navigation devices to confirm the realism of the simulation software.

Figure 14 is actually a copy of the **SDKFNAV CTRL** file for a representative one hour Kalman filter run, using the **SDKFNAV** software package, in order to demonstrate the theoretical capabilities of the filter. For this particular run, filter specifications of the error statistics for the various navaids (shown in the listing of **SDKFNAV CTRL** - i.e. Fig. 14) were matched closely to the actual error statistics being used to generate the simulated navigation data. When there is a perfect match between actual sensor error statistics and the Kalman filter specification of the same parameters, the filter is ideally 'tuned' and should exhibit optimal performance. Figures 15 and 16 show one hour duration time history plots of the two different measurement types (a total of 5 scalar measurements available at each filter update interval). Again, these measurement process time histories, based on simulated navigation data, have been compared with those for representative real navigation data to confirm their accuracy. The high frequency portion of each scalar measurement time history should, and does, conform to the Kalman filter specification of the 1-sigma level of ZMWG measurement noise for the associated measurement component.

TABLE 4.1

ERROR SPECIFICATIONS FOR LTN-90-100 IRS SIMULATION

Initial Position Errors:	Δ LAT = 0.00167 °	ΔLONG = 0.00833 °	Δ ALT = 3.3 Ft
Initial Velocity Errors:	Δv_N = 0.33 Ft/S	Δv_E = 0.33 Ft/S	Δv_z = 0.33 Ft/S
Initial Attitude Errors:	Δ ROLL = 0.012 °	ΔPCH = 0.012 °	Δ HDG = 0.0167 °
Gyro Biases:	0.01 Deg/Hr	Gyro Scale Factors:	5×10^{-6}
Gyro Drifts:	0.003 Deg/Hr	Correlation Times:	3600 Sec
Gyro Random Noise Components:		0.001 Deg/Hr (1σ)	
Accel Biases:	50 μG	Accel Scale Factors:	5×10^{-5}
Accel Drifts:	5.0 μG	Correlation Times:	7200 Sec
Accel Random Noise Components:		5.0 μ G (1σ)	

Baro Scale Factor: 0.05 Correl Time: 5000 Sec Random Noise: 0.1 m (1σ)

TABLE 4.2

ERROR SPECIFICATIONS FOR LORAN-C AND DOPPLER SIMULATIONS

LORAN-C STATISTICS:

Latitude Offset: 0.1 Nm	Latitude Markov: 0.2 Nm	Correl Time: 7200 Sec
Longitude Offset: 0.1 Nm	Longitude Markov: 0.2 Nm	Correl Time: 7200 Sec
Latitude ZMWG Noise: 0.05 Nm (1σ)	Longitude ZMWG Noise: 0.05 Nm (1σ)	

DOPPLER STATISTICS:

SF_U Offset: 0.01	SF_U Markov: 0.01	SF_U Correl Time: 15,000 Sec
BOR_V Offset: 0.5 Deg	BOR_V Markov: 0.65 Deg	BOR_V Correl Time: 15,000 Sec
BOR_W Offset: 0.5 Deg	BOR_W Markov: 0.65 Deg	BOR_W Correl Time: 15,000 Sec
SB_N Offset: 2.0 Ft/S	SB_N Markov: 4.5 Ft/S	SB_N Correl Time: 900 Sec
SB_E Offset: 2.0 Ft/S	SB_E Markov: 4.5 Ft/S	SB_E Correl Time: 900 Sec
U Ch Noise: 5.34 Ft/S (1σ)	V Ch Noise: 5.34 Ft/S (1σ)	W Ch Noise: 5.34 Ft/S (1σ)

Figures 17 to 23 show time history plots that demonstrate the Kalman filter's accuracy in estimating various sets of the error states being modelled. The +/- 1-sigma bounds, as computed by the filter, are also displayed along with the associated filter state error to indicate the level of confidence that the filter is assigning to that particular state estimate. According to the theory of Kalman filtering, when the filter is properly 'tuned' the actual state estimate errors should be within the +/- 1-sigma bounds approximately 67 % of the time. As can be seen from these plots (and results from many other simulation runs as well), this is generally the case for this particular filter run and confirms that the Kalman filter software is fundamentally sound. It should be noted that this Kalman filter run is of short duration compared to the length of a typical operational flight and there is always an initial filter 'transient' time period that shows up in the filter state estimates while the filter is settling down to a steady state condition. Plots of the estimates of IRS sensor errors (i.e. the slowly varying errors in the gyros and accelerometers) show a +/- 1-sigma bound tracking along with the actual filter state estimate (rather than a +/- 1-sigma bound about zero) to give an idea of the uncertainty of that particular estimate. This is the normal graphical display option when the true error state is not available for reference purposes (as would be the case, normally, for real data).

The final plots to be shown, based on this Kalman filter run (Figs. 24 and 25), relate to the so-called residual sequence (or innovations sequence - see Ref. 8) for each of the 5 measurement time series that have already been shown in Figs. 15 and 16. For a given scalar measurement time series $z_i(k)$ (i = 1,..,5), the innovations time series process associated with it can be expressed as

$$\upsilon_i(k) = z_i(k) - H_i(k)\ \delta x'(k) \qquad (20)$$

If the filter is optimally tuned, then Kalman filter theory states that this innovations process will be ZMWG and will have the theoretical statistical variance computed by the filter as

$$p\{\upsilon_i(k)\} = H_i(k)\ P'(k)\ H_i^T(k) + r_i(k) \qquad (21)$$

Figures 24 and 25 indeed show that the innovations time series sequences associated with each of the five scalar measurements being processed by the filter (i.e. both the position and velocity-based measurements) are very close to ZMWG random processes and, furthermore, the +/- 1-sigma bounds being computed (shown in dashed lines superimposed on the same plots as the innovations) appear to closely approximate the true standard deviation in each case. This is clear evidence that the Kalman filter software package is performing properly. For the case of real navigation data, monitoring the innovations sequences during the operation of the Kalman filter can be an effective way of detecting sudden changes in the noise characteristics of the measurement processes that might be due to a sudden fault in one of the measurement sensors (see Refs. 16, 17).

4.5 Conclusions from the Simulation Studies

Prior to working with real navigation data from the Twin Otter, a great many Monte Carlo simulation runs of the Kalman filter were conducted under various simulated conditions, primarily to check out the expected robustness and accuracy of the filter. The following points summarize the results of the analysis of the simulation runs:

a) For this particular Kalman filter configuration, simulation results indicate that there appears to be a small improvement to be gained by using measurement prefiltering on both the Loran-C and Doppler-based measurements, as theory suggests. However, this improvement is not considered sufficient to warrant using prefiltering in a real-time application.

b) There is virtually no difference in filter performance when comparing results with and without the Doppler-based measurements being included. This suggests that, in the presence of the reasonably accurate Loran-C position-based measurements, the rather inaccurate Doppler velocity measurements contribute very little to the estimation of the various IRS error states.

c) One robustness check involved using only an 8 Hz version of the IRS digital data, rather than the usual 16 Hz version. A comparison of Kalman filter runs for these two different IRS data rates confirmed that there was virtually no difference in filter performance.

d) Another filter robustness check consisted of comparisons of filter performance for update intervals longer than the usual 10 seconds. A Kalman filter update interval of 20 seconds provided results almost identical to those for the standard 10 second interval. For a 30 second update interval, a significant degradation in filter performance was apparent.

e) The concept of using so-called Visual On-Tops (VOT's) as infrequent measurements, whenever they occur, was emulated in the filter simulation software in the manner described in Ref. 3. Simulation results confirm that VOT processing is being done properly within the filter software, but some care must be taken as to the details of how VOT measurements are processed in order to avoid filter transients.

The foregoing robustness checks have confirmed the inherent soundness of the Kalman filter design and its ability to degrade gracefully in the presence of computer processing shortcuts that might be required for a real-time version of the filter. By using the simulation software in this manner, various tradeoffs can be made between the ultimate in filter accuracy (with its associated computational complexity) and acceptable accuracy (with significant simplifications in the filtering algorithms).

5.0 KALMAN FILTER RESULTS USING REAL FLIGHT DATA

5.1 Specialized Navigation Flights

Two sets of specialized navigation flight test data (each approximately 1.5 hours in duration) were collected onboard the Twin Otter for use in evaluating and fine tuning the Kalman filter. For these two flights, a route was chosen such that the aircraft overflew selected landmarks that were to be used as reference positions, or Visual On-Tops (VOT's). This provided an assessment of IRS position error drift, Loran-C navigational accuracy, and Kalman filter performance (in regards to horizontal positioning accuracy, at least) for the entire flight. As part of each of these flights, a wind box and/or straight line runs on reciprocal headings along a railroad line were included in order to evaluate wind computation accuracy when filter-corrected IRS velocity components were being used in the wind equations (see Ref. 18 for details of this aspect of the development work). Figure 26 shows the horizontal track plot for the second of these flights (Flight #2) - the numbers from 1 to 15 correspond to the locations of the VOT's along the route (some VOT's were overflown more than once during a flight). The first flight (Flight #1) followed a trajectory identical to the one shown in Fig. 26, except that the wind box was omitted. Table 5.1 lists the set of ten individual landmarks that were used as VOT's, together with their geographical locations which are felt to be accurate to better than 0.1 nm RMS in each of their latitude and longitude components.

TABLE 5.1

GEOGRAPHICAL LOCATIONS OF THE VISUAL ON-TOPS

	LANDMARK DESCRIPTION	LATITUDE	LONGITUDE
1)	RUNWAY INTERS. - OTTAWA AIRPORT	45.31667°	75.66500°
2)	OTTAWA VOR	45.44167°	75.89667°
3)	FITZROY DAM	45.47167°	76.23333°
4)	CHENAUX DAM	45.58333°	76.67500°
5)	MOUNT ST. PATRICK TOWER	45.31000°	76.89167°
6)	MOUNTAIN CHUTE DAM	45.19667°	76.90833°
7)	BARRET CHUTE DAM	45.24333°	76.75833°
8)	BURNSTOWN BRIDGE	45.38500°	75.57833°
9)	WEST END - RAILROAD RUN	45.07667°	75.90333°
10)	EAST END - RAILROAD RUN	45.16000°	75.83833°

5.2 Fundamental Filter Performance Using Real Flight Data

5.2.1 Data from Flight #1 (May '88)

Navigation data from Flight #1 were the first data from the Twin Otter to be analyzed, in detail, from the point of view of suitability for use in the Kalman filter algorithms that had been developed. Initial analysis and use of these data in the Kalman filter program revealed a few minor problems that had to be overcome. The plots of the two position measurement components (i.e. z_1 and z_2, based on the Loran-C position data being differenced with the IRS position data), shown in Fig. 27, indicate two of these problems. A very obvious 'spiking' phenomenon occurs in the time series plot of the longitude measurement component. This effect was traced to a fundamental flaw in the onboard software for formatting the data to be recorded that caused an intermediate-level bit in the IRS longitude digital word to stick at certain times - this software bug was fixed for subsequent flights. In the case of this corrupted set of data, a simple on-line algorithm was used in order to detect and smooth out the spikes prior to the data being used in the Kalman filter. Notice, also, that in the time series plots for each of the measurement components a rather sudden shift of about 0.3 nm occurs several times throughout the flight. A thorough analysis of this phenomenon revealed that the measurement shifts always occurred during times when the aircraft was turning. At first it was thought that the Loran-C receiver might be sensitive to aircraft manoeuvres and build up an error during turns. However, experience using this particular ARNAV Loran-C receiver onboard both the Twin Otter and the Convair 580 has revealed the existence of a consistent 4 second time lag in the output from the receiver that becomes particularly noticeable during manoeuvres. This was corroborated by time-shifting the Loran-C data, relative to the IRS data, by exactly 4 seconds - the spurious shifts in the position measurement data disappeared. Figure 28 shows the same measurement data after applying the corrections that have been described above - each of the time series traces is now relatively smooth, with only the near-constant level of high frequency measurement noise (estimated to be about 0.07 nm RMS for each component) now remaining. Figure 29 shows the corresponding time series traces for the three velocity measurements (i.e. z_3 to z_5, based on the Doppler velocity sensing data being differenced with the IRS velocity data). As can be seen in Fig. 29, Doppler-based measurements are quite noisy, due to the rather high levels of fluctuation noise inherent to this particular Doppler velocity sensor. Moreover, the levels of the fluctuation noise are somewhat different for each of the U, V and W velocity components - i.e. about 3.0 ft/s RMS for U, 10.0 ft/s RMS for V and 5.0 ft/s RMS for W.

The ten VOT's listed in Table 5.1 were used to determine the long-term positioning accuracy of both the LTN-90-100 IRS and Loran-C receiver during the course of the flight. Figure 30 is a set of plots depicting the buildup in IRS positional error (latitude and longitude components) during the course of Flight #1, as determined by position comparisons at the visual landmark locations. Positional error drifts of about 1 nm/hour are typical for this quality of IRS, together with a Schuler-induced oscillating error (with an 84 minute period) of perhaps 1 nm peak-to-peak. There is evidence of a very significant Schuler error component building up in the latitude error trace of Fig. 30, for example. In contrast, Fig. 31 shows a similar series of plots for the Loran-C position errors - note the small, bounded nature of these errors, with both components consistently below 0.2 nm in magnitude. These error levels are consistent with Loran-C system specifications for good coverage areas, such as the area where these data were collected. The small, bounded nature of Loran-C errors makes airborne Loran-C an ideal redundant navaid for identifying the dominant IRS errors using a Kalman filter integrated navigation scheme.

Figure 32 is a listing of the **SDKFNAV CTRL** file used for controlling the running of the **SDKFNAV** software package on this particular set of flight data. The statistical specifications in this file have been fine tuned for the known noise levels of each of the measurement types and for a slightly out-of-specification situation regarding the x accelerometer of the IRS (identified by the manufacturer during a subsequent re-calibration exercise). Also, the VOT lat/long position values have been time shifted so that they are precisely synchronized with the 10 second update intervals of the Kalman filter. Figures 33 to 39 show the time series plots of the complete set of error states estimated by the Kalman filter based on the

control conditions specified in **SDKFNAV CTRL** (i.e. Fig. 32). For the filter run depicted in these plots, the specified measurement noise levels for each of the Loran-based and Doppler-based measurement types were fine-tuned to produce the best overall results. These plots also include dashed lines that indicate the $\pm 1\sigma$ bounds computed by the Kalman filter to reflect the accuracy level of each error state estimate. Note that the error states associated with the vertical channel (i.e. IRS altitude, IRS vertical velocity, IRS vertical acceleration and barometric altimeter bias) remain very close to zero because these errors cannot really be observed by the Kalman filter in the absence of an accurate, independent source of altitude information to use as a redundant measurement type. As well, the sea bias components are zeroed out because no overwater flying occurred during the course of Flight #1. Figure 40 is a plot of the Kalman filter's IRS position calibration accuracy at each of the VOT's occurring during the flight, together with $\pm 1\sigma$ bounds to describe the accuracy predicted by the filter. The uncertainty bounds settle out at approximately $\pm$ 0.2 nm, and the true error of the estimates always falls within the bounds - a good indication of a robust Kalman filter design. Time series plots of the innovations sequences corresponding to each of the 5 scalar measurements (shown in Figs. 40a and 40b) appear to be close to ZMWG random sequences, which is another confirmation of proper Kalman filter performance. Although there is no independent, accurate measure of aircraft inertial velocity to use in order to verify the filter's estimation accuracy for IRS velocity errors, it should be noted that the filter estimates of IRS velocity errors are reasonably consistent when compared to the slope of the position error traces of Fig. 23 (these traces are known to be accurate to within $\pm$0.2 nm). All of the other error states show consistent behaviour and attain values in line with those predicted during simulation studies.

5.2.2 Data from Flight #2 (May '89)

Figure 41 shows the time series plots of the 2 Kalman filter position measurement components (i.e. z_1 and z_2) that are based on the navigation data from Flight #2. Note that, as in the case with Flight #1, the Loran-C data has been time-shifted by 4 seconds in order to synchronize it with the IRS data. The near-constant high frequency measurement noise levels are about the same as for the corresponding data from Flight #1 (see Fig. 28), namely 0.07 nm RMS for each of the components. Figure 42 shows the time series plots for the 3 Kalman filter velocity measurement components (i.e. z_3 to z_5) - the U, V, and W Doppler-based velocity measurements look much the same as the corresponding plots for the Flight #1 data (i.e. Fig. 29) and have the same levels of high frequency measurement noise as on the earlier Flight #1. With regard to the Doppler system, a very minor problem occurred with it during the flight - it went into memory for about 20 seconds early on, due to flying over smooth water for a brief period of time. The simple solution to this problem was to incorporate logic in the Kalman filter program to skip the processing of Doppler-based measurement data around the point in time when the Doppler data were suspect. Figure 43 shows the IRS horizontal position errors at the VOT reference positions during the flight. IRS position error drifting was not quite as bad as what had occurred during Flight #1, due to the fact that an out-of-spec accelerometer had now been repaired. For the same set of VOT reference landmarks, Fig. 44 shows the Loran-C position errors during the course of the flight , with results that are much the same as those for Flight #1 - i.e. the errors in the Loran-C position components are seen to be bounded in magnitude by 0.2 nm.

Figures 45 to 51 show the complete set of error state time histories and associated $\pm 1\sigma$ bounds for a Kalman filter run using the Flight #2 data and an **SDNAV CTRL** file almost identical to the one shown in Fig. 32 - the only difference being that the statistics for the x accelerometer have been changed to reflect the fact that this accelerometer has been re-calibrated and is back within specification. The overall results are quite similar to those for the corresponding filter run on the data from Flight #1 (i.e. Figs. 33 to 39) apart from the obvious fact that IRS horizontal position and velocity errors are now a lot smaller. Finally, Fig. 52 shows the filter's IRS horizontal position calibration accuracy at each of the VOT landmark points during the flight, together with the associated $\pm 1\sigma$ uncertainty bounds that confirm that actual error levels always fall within the bounds, as desired.

5.3 A Comparison of Three Different Measurement Scenarios

There are three different measurement scenarios that can be analyzed easily, based on different combinations of the navaid data being used as Kalman filter measurements. The three configurations to be considered are the following: i) only the Loran-C position measurements available, ii) both the Loran-C and Doppler-based measurements available and iii) only the Doppler-based velocity measurements available. Kalman filter IRS position error estimation results for each of the three filter configurations, and both sets of navigation data, are shown in Figs. 53 to 56. A comparison of position error estimation for each of the three filter configurations reveals that case i) and case ii) results are almost identical, with position errors consistently below 0.2 nm in magnitude (as indicated by the uncertainty bounds), while case iii) results (i.e. Doppler-only) degrade to about 0.6 nm RMS. Kalman filter IRS velocity error estimation results for each of the three filter configurations, and both sets of flight data, are shown in Figs. 57 to 60. The case i) results (Loran-C only) are the most accurate, due to the high quality nature of the Loran-C measurement data. Next in accuracy would be the case ii) results (Loran-C/Doppler measurement combination) - there is a bit of accuracy degradation due to the fact that the inferior quality of the Doppler velocity data makes accurate Doppler error modelling somewhat difficult and slightly corrupts the estimation results, but the error traces are fundamentally the same as in case i). Also, in both of these cases, the RMS error bounds settle out at about 1.0 ft/s (0.3 m/s) for each component. Finally, the case iii) (Doppler only) results are the least accurate, as judged by the wider RMS bounds ($\pm$ 2.3 ft/s or $\pm$ 0.7 m/s on average) and sensitivity to aircraft manoeuvres. However, even in the Doppler-only measurement case, the dominant trends in the IRS velocity error traces of case i) are retained. The foregoing comparisons serve to point out that, even in the absence of the Loran-C position data, there is still a significant improvement in inertial position and velocity accuracy to be gained from the implementation of this Kalman filter design using Doppler-only measurements.

5.4 Conclusions from the Analysis of Real Flight Data

1) Loran-C coverage in the vicinity of Ottawa (i.e. using the NE USA chain) is excellent, with actual Loran-C position errors well within the stated Loran-C performance specification of 0.25 nm RMS for each component. As a result, with Loran-C measurements the Kalman filter can consistently provide IRS-based positioning accuracy to a level of 0.2 nm RMS (i.e. the same accuracy level as Loran-C) and IRS-based velocity accuracy to a level of 1.0 ft/s RMS in the horizontal components. IRS attitude errors can also be decreased somewhat, from 0.05 deg RMS down to 0.03 deg RMS in pitch/roll and from 0.2 deg RMS down to 0.08 deg RMS in heading.

2) With the vertical channel of the IRS being controlled by an internal, third-order baro-damping loop, vertical position accuracy is determined by the fundamental barometric altimeter accuracy, and IRS vertical velocity accuracy is better than 0.5 ft/s RMS without any Kalman filtering being applied. In the absence of a redundant source of altitude information, substantially more accurate than barometric altitude, there is no further improvement to IRS vertical channel accuracy that can be provided by the present version of the Kalman filter. However, the present Kalman filter design is readily adaptable if and when redundant vertical position and/or velocity measurements become available (as would be the case if GPS were included in the suite of navigation sensors).

3) Initial attempts at applying Kalman-corrected IRS inertial velocity data to the wind computation task have proved encouraging (see Ref. 18 for more details). Wind calculations using corrected IRS velocities appear to give more consistent results, and wind computation accuracy of 1 ft/s RMS per channel should be attainable.

4) It has been established that there is no further improvement in IRS error estimation accuracy to be gained by processing the Doppler measurements along with the Loran-C measurements. As a matter of fact, with this particular Doppler velocity sensor, one must be careful to establish the measurement noise 'weighting' properly so as to avoid having the Doppler-based measurement data actually degrade filter performance when it is used in conjunction with the Loran-C data. A Kalman filter analysis of the two sets of specialized navigation flight test data has revealed that Doppler U channel scale factor errors can be 1.5% --> 2%, even after careful calibration of the system, and are quite variable because of the terrain sensitivity of this rather old Doppler set. The V channel boresight error is not particularly constant either, with variations that that can exceed 1 deg at times. With the 200 ft/s groundspeed typically attained by the Twin Otter, uncalibrated Doppler velocity errors in each of the U and V channels can be 3.0 ft/s or more. However, it is worthwhile to retain Doppler velocity measurements in the Kalman filter configuration for situations in which Loran-C coverage is intermittent or simply not available. For the Doppler-only measurement case, IRS horizontal position accuracy is approximately 0.6 nm RMS and horizontal velocity accuracy is about 2.3 ft/s RMS. Due to the error levels inherent to the Doppler data, IRS attitude errors cannot be reduced by any significant amount when only Doppler data are available.

5) The IRS gyro and accelerometer bias-like error states require a much longer time period than 1.5 hours to be estimated accurately by the Kalman filter - i.e. more like 3 to 4 hours (the equivalent of a few periods of the 84 minute Schuler oscillation). However, even in the short term, IRS sensor error state estimation does appear to be yielding consistent intermediate values, judging by the nature of the time histories for these errors over the first 1.5 hours.

6) Analysis of various Kalman filter runs, using a measurement prefiltering option for both the Loran-C and Doppler data, has revealed that estimation accuracy does not improve significantly. It is concluded that measurement prefiltering is not worth the extra computational burden for on-line filtering applications, but should be retained as an option for off-line Kalman filter analysis. Various results using simulated data support this same conclusion.

7) The Kalman filter performed almost identically when using an 8 Hz version of the IRS data, rather than the usual 16 Hz version. This is encouraging from the standpoint of designing a computationally efficient version of the filter for real-time operations. Further experimentation revealed that it is quite feasible to 'slow down' the filter to a 0.05 Hz update rate (i.e. 20 second update interval) without a significant degradation in performance.

8) Under certain circumstances, using VOT's as extra measurements (whenever they occur) can cause an undesirable transient in several of the Kalman filter error states. The process of utilizing VOT's as measurements should be checked out very carefully, for any given Kalman filter configuration, to ensure that basic filtering integrity is maintained.

6.0 CONCLUSIONS AND FUTURE WORK

The Kalman filter integrated navigation design, described in detail in this report, has been demonstrated to meet the performance specifications, when using real flight data, that had been determined from simulation studies. Most importantly, filter-corrected IRS position and velocity components have been verified to be consistently within their predicted accuracy limits. In particular, for the Twin Otter wind computation task, a guaranteed IRS velocity error limit of 1 ft/s will be a very significant improvement in performance compared to the present error limits of 3 to 4 ft/s for Doppler velocity and 3 to 6 ft/s for raw IRS velocity. Applying this Kalman filter approach to airborne wind measurement will ensure that the dominant source of error will no longer be the sensing of inertial velocity but, rather, the measurement of the air data information. Any inconsistencies and errors in wind data that show up during

analysis can then be attributed solely to the air data system and appropriate action taken. The Kalman filter software described in this report has been successfully ported to the VAX/VMS environment. Recent flight data (i.e. from the summer of 1990), acquired using the latest version of the Twin Otter's data acquisition software, has been validated to be completely compatible with the VMS Kalman filter software package. The capability of providing routine Kalman filter-based IRS corrections as a standard feature of the Twin Otter playback software is ready to be tested out on the latest data from Twin Otter field experiments.

Future work will involve confirming the accuracy and reliability of the IRS correction software, using various sets of navigation data from Twin Otter field experiments, by clearly demonstrating consistent improvement in the wind computations that are based on Kalman filter-corrected IRS velocity data. Software will be developed for a Kalman filter/smoother algorithm that is, potentially, more accurate than the present Kalman filter algorithm for post-processing Twin Otter data. A more powerful airborne microprocessor (a DEC MicroVax-II) will soon be installed onboard the Twin Otter. Once that occurs, the Kalman filter software will be installed in it and employed for real-time wind compuations. As it so happens, the present version of the ground-based Kalman filter software can be easily employed for the real-time task. At some point in the near future, the feasibility of installing a GPS receiver onboard the Twin Otter will be assessed with a view to gaining some experience with an IRS/GPS Kalman filter configuration. This sensor combination has the potential for providing inertial data, at a 16 Hz rate, with a real-time absolute accuracy of 30 feet in each position component, 0.3 ft/s in each velocity component, 0.005 deg in each of pitch/roll attitude and 0.02 deg in heading (see Refs. 19, 20). For certain contingency situations that might occur while running a Kalman filter, a careful study will be conducted into the best way to handle unwanted transient errors in a practical manner. Two common contingency situations in which this transient phenomenon can manifest itself are i) the occurrence of a VOT landmark along the flight path, to be processed as an auxiliary position measurement and ii) the case where one of the redundant sources of filter measurement data drops off line at some point during the filter run. One solution to this filter re-configuration problem that is being looked at very seriously is to run several Kalman filters in parallel, each one depicting a possible Kalman filter configuration based on the various sources of navigation data that might be available. The outputs from this bank of Kalman filters would be blended together, using weightings that would depend on real-time information about the integrity of the various navigation sensors that comprise the complete navigation suite.

7.0 ACKNOWLEDGEMENTS

The author would like to thank NRC summer students Elizabeth Crépeau, Eva Finklestein, Patrick Boulé and Tom Fejes for their dedication and hard work in developing and running much of the software used in this research and development program. Thanks are also due to Flight Research staff members Ian MacPherson, John Croll, Ken Lum and Derek Carter (recently retired) for assistance in acquiring Twin Otter flight data and developing a VMS environment for the Kalman filter software. This research was partially funded by the Department of National Defence (DND) and the Atmospheric Environment Service (AES).

8.0 REFERENCES

1. Leach, B.W. — A Kalman Filter Approach to Navigation on the NAE Convair 580 Aeromagnetics Research Aircraft. NAE Report LR-604, National Research Council of Canada, February 1981.

2. Leach, B.W. Maskell, C.A. — VLF Data for Aircraft Navigation Based on an Extended Kalman Filter Design. Proceedings of the Sixth IFAC Symposium on Identification and System Parameter Estimation, Washington, D.C., June 1982.

3. Leach, B.W. — Error State Feedback Kalman Filter Designs for an Airborne Doppler/Heading System. NAE Report LTR-FR-100, National Research Council of Canada, January 1987.

4. Gelb, A. (editor) — Applied Optimal Estimation. The M.I.T. Press, Cambridge, Massachussetts, 1974.

5. Sorenson, H.W. (editor) — Kalman Filtering: Theory and Application. IEEE Press, New York, 1985.

6. Leach, B.W. — An Introduction to Kalman Filtering. NAE Miscellaneous Report 57, National Research Council of Canada, March 1984.

7. Bierman, G.J. — Factorization Methods for Discrete Sequential Estimation. Academic Press, New York, 1977.

8. Maybeck, P.S. — Stochastic Models, Estimation, and Control Volume 1. Academic Press, New York, 1979.

9. — Technical Description of LTN-90-100 Inertial Reference System. Litton Aero Products, Canoga Park, Calif., May 1984.

10. — Decca Overhaul Manual, Doppler Type 70 Series. The Decca Navigator Co. Ltd., Surrey, England, September 1973.

11. — R-40 (AVA-1000A) Operation Manual. ARNAV Systems, Inc., Salem, Oregon, February 1985.

12. MacKenzie, F.D. — Certification of an Airborne Loran-C Navigation System. Navigation, Vol. 29, No. 1, Spring 1982, pp. 69-79.

13. Savage, P.G. — Strapdown System Algorithms. AGARD Lecture Series No. 133 - Advances in Strapdown Inertial Systems, AGARD-LS-133, May 1984, pp. 3-3 to 3-30.

14. Britting, K.R. — Inertial Navigation Systems Analysis. John Wiley & Sons, Inc., Toronto, 1971.

15. Widnall, W.S. Grundy, P.A. — Inertial Navigation System Error Models. Intermetrics Incorporated Report TR-03-73, Cambridge, Massachussetts, May 1973.

16. Maybeck, P.S. — Stochastic Models, Estimation, and Control Volume 2. Academic Press, New York, 1979.

17. Friedland, B.
Grabousky, S.M.
Estimating Sudden Changes of Bias in Linear Dynamic Systems. IEEE Trans. Auto. Control, Vol. AC-27, No. 1, February 1982, pp. 237-240.

18. Leach, B.W.
MacPherson,J.I.
An Improved Airborne Wind Measurement Technique for the NAE Twin Otter. NAE Report NAE-AN-61, National Research Council of Canada, November 1989.

19. DiFilippo, D.J.
Leach, B.W.
A Precise Flight Reference System for Evaluating Airborne Navigation Aids. CASI Journal, Vol. 35, No. 3, September 1989.

20. Denaro, R.P.
Geier, G.J.
GPS/Inertial Navigation System Integration for Enhanced Navigation Performance and Robustness. The NAVSTAR GPS System. AGARD-LS-161, September 1988, pp. 7-1 to 7-12.

APPENDIX A

Fundamental Inertial Navigation Definitions and Equations

The basic inertial system parameters and equations are defined in this Appendix as background material for the details of INS/IRS error modelling and associated Kalman filter design that follow in subsequent Appendices.

a) Model of Earth's Geoid -

The earth's reference ellipsoid (Ref. 14) can be defined in terms of the following three parameters:

R_o (equatorial radius): 3441.726 nm, 6378.165 km or 2.0925732×10^7 ft

e (earth's flattening or ellipticity): 1/298.25 = 0.0033529

ε^2 (earth's eccentricity): $\varepsilon^2 = 2e - e^2 = 0.00669460$

Define L to be geographic latitude and λ to be geographic longitude, with north latitude and west longitude being the positive senses (note that, usually, east longitude is considered positive). Two important radii of curvature are of interest, namely:

r_L (sometimes denoted ρ_m): the meridian radius of curvature

r_λ (sometimes denoted ρ_p): the prime radius of curvature

Approximate expressions for these radii, suitable for INS work, are given as:

$$r_L = R_o [\, 1 - \varepsilon^2 + 1.5\, \varepsilon^2 \sin^2 L \,]$$

$$r_\lambda = R_o [\, 1 + (\varepsilon^2/2) \sin^2 L \,] \qquad \text{(A1)}$$

b) Model of Earth's Gravity -

Let h represent altitude (in feet) above the reference ellipsoid. An adequate mathematical model for so-called plumb bob gravity g_D (i.e. including earth's rotation effects) is given by

$$g_D = G_1(1 - 2h/R_o + 2e \sin^2 L) + G_2(1 - 3 \sin^2 L) - G_3(1 - e \sin^2 L + h/R_o) \cos^2 L \qquad \text{(A2)}$$

with $G_1 = 32.14647$ ft/s^2, $G_2 = 0.0521882$ ft/s^2 and $G_3 = 0.1106653$ ft/s^2.

c) Velocity and Position Equations in Geographic Coordinates -

We want to express aircraft velocity and position components, relative to the rotating earth, using the local-level geographic coordinate frame (i.e. N, E and z) with z positive upwards. First of all, several vectors expressed in the geographic coordinate frame must be defined:

$\mathbf{v}\ (v_N, v_E, v_z)$ - aircraft velocity relative to the earth

$\boldsymbol{\rho}\ (\rho_N, \rho_E, \rho_z)$ - angular velocity of geographic frame relative to the earth

$\mathbf{f}\ (f_N, f_E, f_z)$ - specific force vector (i.e. net acceleration other than that due to gravity)

$\mathbf{g}\ (g_N, g_E, g_z)$ - plumb bob gravity vector

$\boldsymbol{\Omega}\ (\Omega_N, \Omega_E, \Omega_z)$ - angular velocity of the earth relative to inertial space

$\boldsymbol{\omega}(\omega_N, \omega_E, \omega_z)$ - angular velocity of geographic frame relative to inertial space

Let $|\Omega|$ denote the magnitude of the earth rate vector $\boldsymbol{\Omega}$ (the magnitude is 7.2921×10^{-5} rad/s). The components of $\boldsymbol{\Omega}$ are then

$$\Omega_N = |\Omega| \cos L;\ \Omega_E = 0;\ \Omega_z = |\Omega| \sin L \qquad (A3)$$

Let $\boldsymbol{\rho}$ be the angular velocity of the geographic frame relative to the earth which can be expressed in component form as

$$\rho_N = v_E/(r_\lambda + h);\ \rho_E = -v_N/(r_L + h);\ \rho_z = v_E \tan L/(r_\lambda + h) \qquad (A4)$$

Then, the angular velocity of the geographic frame relative to inertial space $\boldsymbol{\omega}$ can be expressed in component form as

$$\omega_N = \rho_N + \Omega_N;\ \omega_E = \rho_E;\ \omega_z = \rho_z + \Omega_z \qquad (A5)$$

It is also useful to note the following expansions:

$$\omega_N + \Omega_N = \rho_N + 2\,\Omega_N = \rho_N + 2\,|\Omega| \cos L$$

$$\omega_z + \Omega_z = \rho_z + 2\,\Omega_z = \rho_z + 2\,|\Omega| \sin L \qquad (A6)$$

With the foregoing definitions, the differential equations describing inertial velocity dynamics are given by

$$\dot{v}_N = f_N + g_N - (\omega_z + \Omega_z) v_E + \omega_E v_z$$

$$\dot{v}_E = f_E + g_E - (\omega_N + \Omega_N) v_z + (\omega_z + \Omega_z) v_N$$

$$\dot{v}_z = f_z + g_z - \omega_E v_N + (\omega_N + \Omega_N) v_E \qquad \text{(A7)}$$

and it is normally assumed that $g_N = g_E = 0$, $g_z = -g_D$. The inertial latitude, longitude and height dynamics are governed by the following differential equations:

$$\dot{L} = v_N / (r_L + h)$$

$$\dot{\lambda} = -v_E / [(r_\lambda + h) \cos L]$$

$$\dot{h} = v_z \qquad \text{(A8)}$$

d) <u>Baro-Inertial Vertical Channel</u> -

It is well known that the vertical navigation equations (i.e. v_z in Eqn. A7 and h in Eqn. A8) are inherently unstable due to the particular form of $g_z = -g_D(h,L)$. A small positive error in h results in a small negative error in g_D and, hence, a small positive error in g_z. This leads to a positive error in v_z, which integrates into a larger and larger error in h - the instability is exponential with a time constant of about ten minutes. This instability is damped out in the LTN-90-100 IRS using a baro altimeter input h_b in a third-order digital filtering loop (see Ref.9). The modified versions of the h and v_z equations to define a third-order loop are

$$\dot{h} = v_z - K_1 (h - h_b)$$

$$\dot{v}_z = f_z + g_z - \omega_E v_N + (\omega_N + \Omega_N) v_E - K_2 (h - h_b) - \delta a$$

$$\dot{\delta a} = K_3 (h - h_b) \qquad \text{(A9)}$$

where δa is a vertical acceleration correction term. A triple pole placement at $s = -1/\tau$ (τ is the time constant of the loop) occurs in the third-order baro-inertial channel if

$$K_1 = 3/\tau;\ K_2 = 3/\tau^2 + 2g/R_o;\ K_3 = 1/\tau^3 \qquad \text{(A10)}$$

where g is the nominal value of earth's gravity. In Litton's LTN-90-100 IRS, τ has been set to 20 seconds, resulting in coefficient values of $K_1 = 0.15$, $K_2 = 0.75 \times 10^{-2}$ and $K_3 = 0.125 \times 10^{-3}$.

e) Direction Cosine Matrix (DCM) -

An important coordinate transformation for the strapdown INS case is the one from aircraft body axes (x, y, z) to local-level geographic coordinates (N, E, z) via the so-called Direction Cosine Matrix (DCM) transformation $\mathbf{C}_b^G$. Aircraft body axes are defined as i) x: longitudinal (forward positive), ii) y: lateral (right positive) and iii) z: normal (down positive). Local-level geographic coordinates are defined as i) N: True North positive, ii) E: True East positive and iii) z: local vertical (up positive). The elements of $\mathbf{C}_b^G$ can be expressed as transcendental functions of the Euler angles (θ, ϕ and ψ) as follows:

$$C_{1,1} = \cos\theta \cos\psi$$

$$C_{1,2} = \sin\theta \sin\phi \cos\psi - \cos\phi \sin\psi$$

$$C_{1,3} = \sin\theta \cos\phi \cos\psi + \sin\phi \sin\psi$$

$$C_{2,1} = \cos\theta \sin\psi$$

$$C_{2,2} = \sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi$$

$$C_{2,3} = \sin\theta \cos\phi \sin\psi - \sin\phi \cos\psi$$

$$C_{3,1} = \sin\theta$$

$$C_{3,2} = -\cos\theta \sin\phi$$

$$C_{3,3} = -\cos\theta \cos\phi \qquad \text{(A11)}$$

APPENDIX B

Error State Equations for the IRS/Doppler/Loran-C Kalman Filter

Details of the error model for the complete 24-state Kalman filter design, in terms of continuous-time differential equations, are given in this Appendix. By explicitly writing out the error state equations in the desired order, the individual elements of matrices **F** and **G** can be readily identified.

a) **Continuous-Time Equations** -

The error state vector/matrix differential equation has the following general form:

$$\dot{\delta x} = \mathbf{F}\,\delta\mathbf{x} + \mathbf{G}\,\mathbf{u} \qquad \text{(B1)}$$

where, in the case of this particular Kalman filter design, the error state vector $\delta\mathbf{x}$ has the 24 elements

$$\delta\mathbf{x} = [\,\delta L\ \delta\lambda\ \delta h\ \delta v_N\ \delta v_E\ \delta v_z\ \varepsilon_N\ \varepsilon_E\ \varepsilon_z\ \delta a\ B\omega_x\ B\omega_y\ B\omega_z\ Ba_x\ Ba_y\ Ba_z\ Bh_b\ B_{LAT}\ B_{LNG}\ SF_U\ B_V\ B_W\ SB_N\ SB_E\,]^T \qquad \text{(B2)}$$

and the ZMWG plant noise vector **u** has the 24 elements

$$\mathbf{u} = [\,u\omega_x\ u\omega_y\ u\omega_z\ ua_x\ ua_y\ ua_z\ ug_N\ ug_E\ ug_z\ uh_b\ uB\omega_x\ uB\omega_y\ uB\omega_z\ uBa_x\ uBa_y\ uBa_z\ uBh_b\ uB_{LT}\ uB_{LG}\ uSF_U\ uB_V\ uB_W\ uSB_N\ uSB_E\,]^T \qquad \text{(B3)}$$

b) **Definition of Error States** -

A very short description of each of the error states is given below:

1. δL - error in IRS latitude (rad)
2. $\delta\lambda$ - error in IRS longitude (rad)
3. δh - error in IRS altitude (feet)
4. δv_N - error in IRS north velocity component (ft/sec)
5. δv_E - error in IRS east velocity component (ft/sec)
6. δv_z - error in IRS local vertical velocity component (ft/sec)
7. ε_N - IRS attitude error about N (north) axis (rad)
8. ε_E - IRS attitude error about E (east) axis (rad)
9. ε_z - IRS attitude error about z (local vertical) axis (rad)

10. δa - acceleration correction in baro damped vertical channel
11. $B\omega_x$ - x axis strapdown gyro Markov error (rad/sec)
12. $B\omega_y$ - y axis strapdown gyro Markov error (rad/sec)
13. $B\omega_z$ - z axis strapdown gyro Markov error (rad/sec)
14. Ba_x - x axis strapdown accelerometer Markov error (ft/sec^2)
15. Ba_y - y axis strapdown accelerometer Markov error (ft/sec^2)
16. Ba_z - z axis strapdown accelerometer Markov error (ft/sec^2)
17. Bh_b - baro altimeter Markov error (feet)
18. B_{LAT} - Loran-C latitude Markov error (rad)
19. B_{LNG} - Loran-C longitude Markov error (rad)
20. SF_U - Doppler U channel scale factor Markov error ( )
21. B_V - Doppler V channel boresight Markov error (rad)
22. B_W - Doppler W channel boresight Markov error (rad)
23. SB_N - North Markov sea bias component (ft/sec)
24. SB_E - East Markov sea bias component (ft/sec)

c) **Definition of Plant Noise Components** -

The 24 elements of the plant noise vector **u** can be described as follows:

1. $u\omega_x$ - x axis gyro random drift (rad/sec)
2. $u\omega_y$ - y axis gyro random drift (rad/sec)
3. $u\omega_z$ - z axis gyro random drift (rad/sec)
4. ua_x - x axis accelerometer random noise (ft/sec^2)
5. ua_y - y axis accelerometer random noise (ft/sec^2)
6. ua_z - z axis accelerometer random noise (ft/sec^2)
7. ug_N - North component of random gravity anomaly (ft/sec^2)
8. ug_E - East component of random gravity anomaly (ft/sec^2)
9. ug_z - vertical component of random gravity anomaly (ft/sec^2)
10. uh_b - baro altimeter random noise (feet)
11. $uB\omega_x$ - x axis gyro Markov driving noise (rad/s$^{3/2}$)
12. $uB\omega_y$ - y axis gyro Markov driving noise (rad/s$^{3/2}$)
13. $uB\omega_z$ - z axis gyro Markov driving noise (rad/s$^{3/2}$)
14. uBa_x - x accelerometer Markov driving noise (ft/s$^{5/2}$)
15. uBa_y - y accelerometer Markov driving noise (ft/s$^{5/2}$)
16. uBa_z - z accelerometer Markov driving noise (ft/s$^{5/2}$)
17. uBh_b - baro altimeter Markov driving noise (ft/s$^{1/2}$)
18. uB_{LT} - Loran-C latitude Markov driving noise (rad/s$^{1/2}$)
19. uB_{LG} - Loran-C longitude Markov driving noise (rad/s$^{1/2}$)
20. uSF_U - Doppler U channel scale factor Markov driving noise ( )
21. uB_V - Doppler V channel boresight Markov driving noise (rad/s$^{1/2}$)
22. uB_W - Doppler W channel boresight Markov driving noise (rad/s$^{1/2}$)
23. uSB_N - North sea bias Markov driving noise (ft/s$^{3/2}$)
24. uSB_E - East sea bias Markov driving noise (ft/s$^{3/2}$)

d) Relationship Between Doppler Velocity Errors and Doppler Error States -

The relationship between the actual errors in each of the U, V and W Doppler channels and the Doppler error states being modelled in the Kalman filter can be expressed very simply when only first-order errors are considered. Define δU, δV and δW to be the errors in the U, V and W Doppler channels respectively. Let SF_U, B_V and B_W be the Doppler U scale factor error, V boresight error and W boresight error respectively. Then the relationships between these two forms of the Doppler errors are as follows:

$$\delta U = SF_U\, U \; ; \; \delta V = B_V\, U \; ; \; \delta W = B_W\, U \qquad \text{(B4)}$$

where SF_U is dimensionless and B_V, B_W are in the dimensions of radians.

e) The 24 Scalar Differential Equations -

For each of the 24 error states, the explicit scalar differential equation is written out with the non-zero terms ordered in such a way that it will be easy to identify the corresponding elements in **F** and **G**.

$$\dot{\delta L} = -(v_N / R_o^2)\,\delta h + (1 / R_o)\,\delta v_N$$

$$\dot{\delta\lambda} = -[v_E \tan L / (R_o \cos L)]\,\delta L + [v_E / (R_o^2 \cos L)]\,\delta h - [1 / (R_o \cos L)]\,\delta v_E$$

$$\dot{\delta h} = -K_1\,\delta h + \delta v_z + K_1 B h_b + K_1\, u h_b$$

$$\begin{aligned}\dot{\delta v}_N = {} & [-2|\Omega| v_E \cos L - v_E^2 / (R_o \cos^2 L)]\,\delta L + [(v_E^2 \tan L + v_N v_z) / R_o^2]\,\delta h - (v_z / R_o)\,\delta v_N \\ & - [2|\Omega| \sin L + 2 v_E \tan L / R_o]\,\delta v_E - (v_N / R_o)\,\delta v_z + a_z \varepsilon_E - a_E \varepsilon_z \\ & + C_{1,1} B a_x + C_{1,2} B a_y + C_{1,3} B a_z + C_{1,1} u a_x + C_{1,2} u a_y + C_{1,3} u a_z + u g_N\end{aligned}$$

$$\begin{aligned}\dot{\delta v}_E = {} & [2|\Omega| (v_N \cos L + v_z \sin L) + v_E v_N / (R_o \cos^2 L)]\,\delta L + [(v_E v_z - v_E v_N \tan L) / R_o^2]\,\delta h \\ & + (2|\Omega| \sin L + v_E \tan L / R_o)\,\delta v_N + (v_N \tan L / R_o - v_z / R_o)\,\delta v_E \\ & - (2|\Omega| \cos L + v_E / R_o)\,\delta v_z - a_z \varepsilon_N + a_N \varepsilon_z + C_{2,1} B a_x + C_{2,2} B a_y + C_{2,3} B a_z \\ & + C_{2,1} u a_x + C_{2,2} u a_y + C_{2,3} u a_z + u g_E\end{aligned}$$

$$\begin{aligned}\dot{\delta v}_z = {} & -2|\Omega| v_E \sin L\,\delta L + [2g / R_o - (v_E^2 + v_N^2) / R_o^2 - K_2]\,\delta h + 2 (v_N / R_o)\,\delta v_N \\ & + (2|\Omega| \cos L + 2 v_E / R_o)\,\delta v_E + a_E \varepsilon_N - a_N \varepsilon_E - \delta a + C_{3,1} B a_x + C_{3,2} B a_y + C_{3,3} B a_z \\ & + K_2 B h_b + C_{3,1} u a_x + C_{3,2} u a_y + C_{3,3} u a_z + u g_z + K_2\, u h_b\end{aligned}$$

$$\begin{aligned}\dot{\varepsilon}_N = {} & -|\Omega| \sin L\,\delta L - v_E / R_o^2\,\delta h + (1 / R_o)\,\delta v_E - (|\Omega| \sin L + v_E \tan L / R_o)\,\varepsilon_E - (v_N / R_o)\varepsilon_z \\ & + C_{1,1} B\omega_x + C_{1,2} B\omega_y + C_{1,3} B\omega_z + C_{1,1} u\omega_x + C_{1,2} u\omega_y + C_{1,3} u\omega_z\end{aligned}$$

$$\begin{aligned}\dot{\varepsilon}_E = {} & (v_N / R_o^2)\,\delta h - (1 / R_o)\,\delta v_N + (|\Omega| \sin L + v_E \tan L / R_o)\,\varepsilon_N - (|\Omega| \cos L + v_E / R_o)\,\varepsilon_z \\ & + C_{2,1} B\omega_x + C_{2,2} B\omega_y + C_{2,3} B\omega_z + C_{2,1} u\omega_x + C_{2,2} u\omega_y + C_{2,3} u\omega_z\end{aligned}$$

$$\begin{aligned}\dot{\varepsilon}_z = {} & [|\Omega| \cos L + v_E / (R_o \cos^2 L)]\,\delta L - v_E (\tan L / R_o^2)\,\delta h + (\tan L / R_o)\,\delta v_E \\ & + (v_N / R_o)\,\varepsilon_N + (|\Omega| \cos L + v_E / R_o)\,\varepsilon_E + C_{3,1} B\omega_x + C_{3,2} B\omega_y + C_{3,3} B\omega_z \\ & + C_{3,1} u\omega_x + C_{3,2} u\omega_y + C_{3,3} B\omega_z\end{aligned}$$

$$\dot{\delta a} = K_3\,\delta h - K_3 B h_b - K_3\, u h_b$$

$$\dot{B\omega}_x = -(1 / \tau_{\omega x})\, B\omega_x + u B\omega_x \; ; \; \dot{B\omega}_y = -(1 / \tau_{\omega y})\, B\omega_y + u B\omega_y \; ; \; \dot{B\omega}_z = -(1 / \tau_{\omega z})\, B\omega_z + u B\omega_z$$

$$\dot{Ba}_x = -(1/\tau_{ax})\,Ba_x + uBa_x;\quad \dot{Ba}_y = -(1/\tau_{ay})\,Ba_y + uBa_y;\quad \dot{Ba}_z = -(1/\tau_{az})\,Ba_z + uBa_z$$

$$\dot{Bh}_b = -(1/\tau_{hb})\,Bh_b + uBh_b;\quad \dot{B}_{LAT} = -(1/\tau_{LT})\,B_{LAT} + uB_{LT};\quad \dot{B}_{LNG} = -(1/\tau_{LG})B_{LNG} + uB_{LG}$$

$$\dot{SF}_U = -(1/\tau_{SF})\,SF_U + uSF_U;\quad \dot{B}_V = -(1/\tau_{BV})\,B_V + uB_V;\quad \dot{B}_W = -(1/\tau_{BW})\,B_W + uB_W$$

$$\dot{SB}_N = -(1/\tau_S)\,SB_N + uSB_N;\quad \dot{SB}_E = -(1/\tau_S)\,SB_E + uSB_E$$

f) **Identification of Non-Zero F_{ij}'s** -

Based on the explicit error state scalar differential equations shown above, the non-zero elements of **F** can be identified as follows:

$$F_{1,3} = -v_N/R_o^2;\quad F_{1,4} = 1/R_o$$

$$F_{2,1} = -v_E \tan L/(R_o \cos L);\quad F_{2,3} = v_E/(R_o^2 \cos L);\quad F_{2,5} = -1/(R_o \cos L)$$

$$F_{3,3} = -K_1;\quad F_{3,6} = 1;\quad F_{3,17} = K_1$$

$$F_{4,1} = -2\,|\Omega|\,v_E \cos L - v_E^2/(R_o \cos^2 L)$$

$$F_{4,3} = (v_E^2 \tan L + v_N v_z)/R_o^2;\quad F_{4,4} = -v_z/R_o$$

$$F_{4,5} = -2\,|\Omega| \sin L - 2\,v_E \tan L/R_o;\quad F_{4,6} = -v_N/R_o$$

$$F_{4,8} = a_z;\quad F_{4,9} = -a_E;\quad F_{4,14} = C_{1,1};\quad F_{4,15} = C_{1,2};\quad F_{4,16} = C_{1,3}$$

$$F_{5,1} = 2\,|\Omega|\,(v_N \cos L + v_z \sin L) + v_E\,v_N/(R_o \cos^2 L)$$

$$F_{5,3} = (v_E\,v_z - v_E\,v_N \tan L)/R_o^2;\quad F_{5,4} = 2\,|\Omega| \sin L + v_E \tan L/R_o$$

$$F_{5,5} = v_N \tan L/R_o - v_z/R_o;\quad F_{5,6} = -2\,|\Omega| \cos L - v_E/R_o;\quad F_{5,7} = -a_z;\quad F_{5,9} = a_N$$

$$F_{5,14} = C_{2,1};\quad F_{5,15} = C_{2,2};\quad F_{5,16} = C_{2,3}$$

$$F_{6,1} = -2\,|\Omega|\,v_E \sin L;\quad F_{6,3} = 2g/R_o - (v_E^2 + v_N^2)/R_o^2 - K_2$$

$$F_{6,4} = 2\,v_N/R_o;\quad F_{6,5} = 2\,|\Omega| \cos L + 2\,v_E/R_o;\quad F_{6,7} = a_E;\quad F_{6,8} = -a_N;\quad F_{6,10} = -1$$

$$F_{6,14} = C_{3,1};\quad F_{6,15} = C_{3,2};\quad F_{6,16} = C_{3,3};\quad F_{6,17} = K_2$$

$$F_{7,1} = -|\Omega| \sin L;\quad F_{7,3} = -v_E/R_o^2;\quad F_{7,5} = 1/R_o;\quad F_{7,8} = -|\Omega| \sin L - v_E \tan L/R_o$$

$$F_{7,9} = -v_N/R_o;\quad F_{7,11} = C_{1,1};\quad F_{7,12} = C_{1,2};\quad F_{7,13} = C_{1,3}$$

$$F_{8,3} = v_N/R_o^2;\quad F_{8,4} = -1/R_o;\quad F_{8,7} = |\Omega| \sin L + v_E \tan L/R_o$$

$F_{8,9} = -|\Omega| \cos L - v_E / R_o$; $F_{8,11} = C_{2,1}$; $F_{8,12} = C_{2,2}$; $F_{8,13} = C_{2,3}$

$F_{9,1} = |\Omega| \cos L + v_E / (R_o \cos^2 L)$; $F_{9,3} = -v_E \tan L / R_o^2$; $F_{9,5} = \tan L / R_o$

$F_{9,7} = v_N / R_o$; $F_{9,8} = |\Omega| \cos L + v_E / R_o$; $F_{9,11} = C_{3,1}$; $F_{9,12} = C_{3,2}$; $F_{9,13} = C_{3,3}$

$F_{10,3} = K_3$; $F_{10,17} = -K_3$

$F_{11,11} = -1 / \tau_{\omega x}$; $F_{12,12} = -1 / \tau_{\omega y}$; $F_{13,13} = -1 / \tau_{\omega z}$

$F_{14,14} = -1 / \tau_{ax}$; $F_{15,15} = -1 / \tau_{ay}$; $F_{16,16} = -1 / \tau_{az}$

$F_{17,17} = -1 / \tau_{hb}$; $F_{18,18} = -1 / \tau_{LT}$; $F_{19,19} = -1 / \tau_{LG}$; $F_{20,20} = -1 / \tau_{SF}$

$F_{21,21} = -1 / \tau_{BV}$; $F_{22,22} = -1 / \tau_{BW}$; $F_{23,23} = -1 / \tau_S$; $F_{24,24} = -1 / \tau_S$

g) **<u>Identification of Non-Zero G_{ij}'s</u>** -

The non-zero elements of **G** can be identified as

$G_{3,10} = K_1$

$G_{4,4} = C_{1,1}$; $G_{4,5} = C_{1,2}$; $G_{4,6} = C_{1,3}$; $G_{4,7} = 1$

$G_{5,4} = C_{2,1}$; $G_{5,5} = C_{2,2}$; $C_{5,6} = C_{2,3}$; $G_{5,8} = 1$

$G_{6,4} = C_{3,1}$; $G_{6,5} = C_{3,2}$; $G_{6,6} = C_{3,3}$; $G_{6,9} = 1$; $G_{6,10} = K_2$

$G_{7,1} = C_{1,1}$; $G_{7,2} = C_{1,2}$; $G_{7,3} = C_{1,3}$

$G_{8,1} = C_{2,1}$; $G_{8,2} = C_{2,2}$; $G_{8,3} = C_{2,3}$

$G_{9,1} = C_{3,1}$; $G_{9,2} = C_{3,2}$; $G_{9,3} = C_{3,3}$

$G_{10,10} = -K_3$; $G_{11,11} = 1$; $G_{12,12} = 1$; $G_{13,13} = 1$; $G_{14,14} = 1$

$G_{15,15} = 1$; $G_{16,16} = 1$; $G_{17,17} = 1$; $G_{18,18} = 1$; $G_{19,19} = 1$

$G_{20,20} = 1$; $G_{21,21} = 1$; $G_{22,22} = 1$; $G_{23,23} = 1$; $G_{24,24} = 1$

APPENDIX C

Kalman Filter Measurement Processing

There are two types of measurements to be processed in the Kalman filter: i) discrete measurements based on position differences between IRS position components and Loran-C position components available every second; and ii) discrete measurements based on velocity differences between IRS velocity components and Doppler radar velocity components available every half second. In this Appendix the measurement error models and observation equations are derived for both measurement types.

a) Measurements Based on Loran-C -

At any discrete time t_k assume that two simultaneous measurements are processed by the Kalman filter based on airborne Loran-C position data, i.e.

$$z_1(k) = z_1(t_k) = LAT_{IRS}(t_k) - LAT_{LOR}(t_k)$$

$$z_2(k) = z_2(t_k) = LNG_{IRS}(t_k) - LNG_{LOR}(t_k) \qquad (C1)$$

where LAT_{IRS}, LNG_{IRS} are the IRS estimates of position and LAT_{LOR}, LNG_{LOR} are the Loran-C estimates of position, for the same point in time t_k. Let $LAT_{Tr}(t_k)$ and $LNG_{Tr}(t_k)$ be the true geographical position of the aircraft at t_k. Define δLAT_{IRS}, δLNG_{IRS} to be the errors associated with the IRS calculation of position and define δLAT_{LOR}, δLNG_{LOR} to be the errors associated with the Loran-C measurement of position. Then $z_1(k)$ and $z_2(k)$ can be expressed as

$$z_1(k) = [LAT_{Tr}(t_k) + \delta LAT_{IRS}(t_k)] - [LAT_{Tr}(t_k) + \delta LAT_{LOR}(t_k)]$$

$$= \delta LAT_{IRS}(t_k) - \delta LAT_{LOR}(t_k)$$

$$z_2(k) = [LNG_{Tr}(t_k) + \delta LNG_{IRS}(t_k)] - [LNG_{Tr}(t_k) + \delta LNG_{LOR}(t_k)]$$

$$= \delta LNG_{IRS}(t_k) - \delta LNG_{LOR}(t_k) \qquad (C2)$$

A suitable error model for the Loran-C position components would consist of a bias-like, first-order Markov process together with additive ZMWG random noise. Thus, we can write

$$\delta LAT_{LOR}(t_k) = B_{LAT}(t_k) + v_{LLT}(t_k)\ ;\ v_{LLT}\ \text{ZMWG}$$

$$\delta LNG_{LOR}(t_k) = B_{LNG}(t_k) + v_{LLG}(t_k) \; ; \; v_{LLG} \; ZMWG \tag{C3}$$

where the discrete-time form of the equations describing the Markov error processes, $B_{LAT}(k)$ and $B_{LNG}(k)$, would be

$$B_{LAT}(k+1) = \exp(-\Delta T/\tau_{LT})\, B_{LAT}(k) + uB_{LT}(k)$$

$$B_{LNG}(k+1) = \exp(-\Delta T/\tau_{LG})\, B_{LNG}(k) + uB_{LG}(k) \tag{C4}$$

with uB_{LT} and uB_{LG} being the ZMWG Markov driving noises associated with these first-order Markov processes. Note that Eqns. C4 are simply the discrete-time equivalents of the corresponding continuous-time Markov equations for Loran-C position error states that have been given in Appendix B.

The general form of the scalar observation equation for the i^{th} measurement at t_k is

$$z_i(k) = \mathbf{H}_i(k)\, \delta\mathbf{x}(k) + v_i(k) \; ; \; v_i \; ZMWG \tag{C5}$$

and we wish to identify the non-zero elements of $\mathbf{H}_1$ and $\mathbf{H}_2$, which will correspond to the Loran-based measurements that are defined in Eqn. C1. The complete error state vector $\delta\mathbf{x}$ has already been defined in Appendix B. If we look at Eqns. C3 in terms of elements of $\delta\mathbf{x}$ we can identify

$$\delta LAT_{IRS} = \delta L \; ; \; \delta LNG_{IRS} = \delta\lambda$$

$$\delta LAT_{LOR} = B_{LAT} + v_{LLT} \; ; \; \delta LNG_{LOR} = B_{LNG} + v_{LLG} \tag{C6}$$

Then Eqns. C2 can be written in the alternative form

$$z_1(k) = \delta L(k) - B_{LAT}(k) - v_{LLT}(k)$$

$$z_2(k) = \delta\lambda(k) - B_{LNG}(k) - v_{LLG}(k) \tag{C7}$$

and we can identify $v_1(k) = -v_{LLT}(k)$, $v_2(k) = -v_{LLG}(k)$; with

$$\mathbf{H}_1(k) = [1\,0\,0\,0 \underset{\#18}{\ldots\ldots\ldots\ldots} 0\,0\,{-1}\,0\,0\,0\,0\,0\,0]$$

$$\mathbf{H}_2(k) = [0\,1\,0\,0 \underset{\#19}{\ldots\ldots\ldots\ldots} 0\,0\,0\,{-1}\,0\,0\,0\,0\,0] \tag{C8}$$

The only non-zero elements of $\mathbf{H}_1$ and $\mathbf{H}_2$ are seen to be $h_{1,1} = 1$, $h_{1,18} = -1$, $h_{2,1} = 1$ and $h_{2,19} = -1$. The ZMWG measurement noise components, v_1 and v_2, are identified as simply the negatives of the corresponding Loran-C ZMWG random noise components.

b) Measurements Based on Doppler Radar -

Assume that the fundamental data available from the Doppler radar system are in the form of a three-component body axis velocity vector $\mathbf{v}^b_D$, with the components denoted as U_D, V_D and W_D (the positive sense being forward, to the right and down respectively). It is then appropriate to process the velocity differences between the IRS and the Doppler radar in the body axis frame. To do this correctly, IRS velocity components have to be transformed into equivalent body axis components and the Doppler velocity data must be corrected for lever arm effects. The lever arm effects are due to the fact that the Doppler radar antenna is in a different location than that of the IRS. In order to express the observation equations properly, we must also develop error equations for each of the IRS and Doppler velocity components in the body axis frame of reference.

Transformation of IRS Velocity Vector into Body Axes:

Let $\mathbf{v}^G_I$ represent the three-component IRS velocity vector in the geographic reference frame, where the individual components are v_N, v_E and v_z. Denote by $\mathbf{C}^b_G$ the transformation matrix that converts $\mathbf{v}^G_I$ into an equivalent velocity vector in body axis coordinates $\mathbf{v}^b_I$, having components U_I, V_I, W_I. In vector/matrix terminology, the transformation equation is simply

$$\mathbf{v}^b_I = \mathbf{C}^b_G \, \mathbf{v}^G_I \qquad \text{(C9)}$$

This particular transformation matrix can be re-computed continuously using Euler angle (i.e. attitude) information from the LTN-90-100 IRS. Note that the transformation $\mathbf{C}^b_G$ is the inverse transformation of the DCM that has already been defined in Appendix A and denoted as $\mathbf{C}^G_b$. For orthogonal reference frames, such as the ones we are dealing with, it can be shown that

$$\mathbf{C}^b_G = (\mathbf{C}^G_b)^{-1} = (\mathbf{C}^G_b)^T \qquad \text{(C10)}$$

Taking the transpose of $\mathbf{C}^G_b$, as defined in Eqns. A11, results in the following elements for $\mathbf{C}^b_G$:

$$\begin{aligned}
C_{1,1}' &= \cos\theta \cos\psi \\
C_{1,2}' &= \cos\theta \sin\psi \\
C_{1,3}' &= \sin\theta \\
C_{2,1}' &= \sin\theta \sin\phi \cos\psi - \cos\phi \sin\psi \\
C_{2,2}' &= \sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi \\
C_{2,3}' &= -\cos\theta \sin\phi \\
C_{3,1}' &= \sin\theta \cos\phi \cos\psi + \sin\phi \sin\psi \\
C_{3,2}' &= \sin\theta \cos\phi \sin\psi - \sin\phi \cos\psi \\
C_{3,3}' &= -\cos\theta \cos\phi
\end{aligned} \qquad \text{(C11)}$$

Based on the foregoing definition of the elements of $\mathbf{C}^{b}_{G}$, the IRS velocity components can be expressed in body axes as

$$U_I = C_{1,1}\cdot v_N + C_{1,2}\cdot v_E + C_{1,3}\cdot v_z$$

$$V_I = C_{2,1}\cdot v_N + C_{2,2}\cdot v_E + C_{2,3}\cdot v_z$$

$$W_I = C_{3,1}\cdot v_N + C_{3,2}\cdot v_E + C_{3,3}\cdot v_z \qquad \text{(C12)}$$

Correction of Doppler Velocities for Lever Arm Effects:

The strapdown IRS and the Doppler radar antenna are not co-located on the Twin Otter aircraft. Due to the length of the lever arm between the two locations, there will be significant relative motion between the two navaids caused by typical aircraft attitude changes. It is thus important to correct the Doppler velocity components for this lever arm effect so that the velocity difference measurements, referenced to the location of the IRS, are not in error. Let $\boldsymbol{l}_{DI}$ represent a three-component lever arm position vector, in body axis coordinates, going <u>from</u> the Doppler radar antenna <u>to</u> the LTN-90-100 IRS. The individual components of $\boldsymbol{l}_{DI}$ will be defined as l_x, l_y, l_z and they have the values l_x = 13.97 ft, l_y = 0.91 ft, l_z = - 1.34 ft. Let $\boldsymbol{\omega}^{b}_{B}$ be the angular body rate vector describing the angular rotation rate of the IRS (and, hence, the aircraft) as expressed in body frame coordinates. The components of $\boldsymbol{\omega}^{b}_{B}$ (i.e. ω_{Bx}, ω_{By}, ω_{Bz}) are measured by the IRS, and would be directly available in digital form from the IRS dataport. Denote the <u>uncorrected</u> Doppler velocity vector by $\mathbf{v}^{b}_{D}$ (with components U_D, V_D, W_D) and the corresponding <u>corrected</u> velocity vector by $\mathbf{v}^{b}_{DC}$ (with components U_{DC}, V_{DC}, W_{DC}). The vector equation that describes the Doppler velocity correction for lever arm effects is then simply

$$\mathbf{v}^{b}_{DC} = \mathbf{v}^{b}_{D} + \boldsymbol{\omega}^{b}_{B} \times \boldsymbol{l}_{DI} \qquad \text{(C13)}$$

The standard vector cross product formula results in

$$\boldsymbol{\omega}^{b}_{B} \times \boldsymbol{l}_{DI} = \begin{bmatrix} 0 & -\omega_{Bz} & \omega_{By} \\ \omega_{Bz} & 0 & -\omega_{Bx} \\ -\omega_{By} & \omega_{Bx} & 0 \end{bmatrix} \begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix}$$

$$= \begin{bmatrix} -\omega_{Bz}\, l_y + \omega_{By}\, l_z \\ \omega_{Bz}\, l_x - \omega_{Bx}\, l_z \\ -\omega_{By}\, l_x + \omega_{Bx}\, l_y \end{bmatrix} \qquad \text{(C14)}$$

The corrected Doppler velocity components can then be identified as

$$U_{DC} = U_D - \omega_{Bz}\, l_y + \omega_{By}\, l_z$$

$$V_{DC} = V_D + \omega_{Bz}\, l_x - \omega_{Bx}\, l_z$$

$$W_{DC} = W_D - \omega_{By}\, l_x + \omega_{Bx}\, l_y \qquad \text{(C15)}$$

Once the IRS velocity components have been transformed into the body-axis coordinate frame (Eqns. C12), and the Doppler velocity components have been corrected for the lever arm effects (Eqns. C15), the three Doppler-based measurement components can then be computed as

$$z_3(k) = U_I(k) - U_{DC}(k)$$

$$z_4(k) = V_I(k) - V_{DC}(k)$$

$$z_5(k) = W_I(k) - W_{DC}(k) \qquad \text{(C16)}$$

at any discrete time t_k.

Error Modelling for Doppler-Based Measurements:

Let $\delta v^b{}_m$ denote the three-component vector of differences between IRS and Doppler velocities as computed in the body-axis frame (the elements of $\delta v^b{}_m$ are exactly the components expressed in Eqns. C16). A vector/matrix equation for $\delta v^b{}_m$ would be

$$\delta v^b{}_m = v^b{}_I - v^b{}_{DC}$$

$$= C^b{}_{GI}\, v^G{}_I - v^b{}_{DC} \qquad \text{(C17)}$$

In order to identify a suitable Kalman filter observation equation, we would like to derive an expression for $\delta v^b{}_m$ in terms of IRS and Doppler error quantities. The errors in $C^b{}_G$, $v^G{}_I$ and $v^b{}_{DC}$ can be expressed as

$$\delta v^G{}_I = v^G{}_I - v^G + (\varepsilon^{GC} \times) v^G$$

$$\delta v^b{}_{DC} = v^b{}_{DC} - v^b$$

$$\delta C^b{}_G = C^b{}_{GI} - C^b{}_G = (\varepsilon^{GC} \times) \qquad \text{(C18)}$$

where

v^G = true velocity vector in geographic coordinates

v^b = true velocity vector in body axis coordinates

$C^b{}_G$ = true transformation from geographic to body axes

$\mathbf{C}^b_{GI}$ = transformation from geographic to body axes as computed from error corrupted IRS attitude data

$(\boldsymbol{\varepsilon}^{GC} \times)$ = skew symmetric matrix expressing misalignment of IRS platform with respect to computer frame (see Refs. 13, 14)

Substituting the expressions of Eqns. C18 into Eqn. C17 results in

$$\delta \mathbf{v}^b_m = \mathbf{C}^b_{GI}\, \delta \mathbf{v}^G_I - \mathbf{C}^b_G (\boldsymbol{\varepsilon}^{GC} \times)(\boldsymbol{\varepsilon}^{GC} \times)\mathbf{v}^G - \delta \mathbf{v}^b_{DC} \tag{C19}$$

which, to first order, becomes

$$\delta \mathbf{v}^b_m = \mathbf{C}^b_{GI}\, \delta \mathbf{v}^G_I - \delta \mathbf{v}^b_{DC} \tag{C20}$$

We can identify the components of $\delta \mathbf{v}_{GI}$ as

$$\delta \mathbf{v}_{GI} = [\delta v_N \; \delta v_E \; \delta v_z]^T \tag{C21}$$

where δv_N, δv_E and δv_z are simply the IRS velocity error states of the Kalman filter design. The components of $\mathbf{C}^b_{GI}$ can be computed from Eqns. C11, where the attitude parameters are supplied by the IRS. It then only remains to define a suitable Doppler error model for $\delta \mathbf{v}^b_{DC}$ so that $\delta \mathbf{v}^b_m$ can be completely specified in terms of error states.

The main contributions to the Doppler velocity error vector $\delta \mathbf{v}^b_{DC}$ will consist of i) scale factor error, ii) boresight error (i.e. Doppler antenna misalignment), iii) high frequency fluctuation noise and iv) ocean current effects (when applicable). For a fixed-wing aircraft, in which it can be assumed that the aircraft velocity vector is approximately aligned with the U (longitudinal) body axis, an adequate error model for $\delta \mathbf{v}^b_{DC}$ is the following:

$$\delta \mathbf{v}^b_{DC} = \begin{bmatrix} \delta U_{DC} \\ \delta V_{DC} \\ \delta W_{DC} \end{bmatrix} = \begin{bmatrix} SF_U \\ B_V \\ B_W \end{bmatrix} U_T + \mathbf{C}^b_{GI} \begin{bmatrix} SB_N \\ SB_E \\ 0 \end{bmatrix} + \begin{bmatrix} n_U \\ n_V \\ n_W \end{bmatrix} \tag{C22}$$

where

SF_U = U channel Doppler scale factor error

B_V, B_W = V, W components of Doppler boresight error

U_T = true U body velocity component

SB_N, SB_E = north, east components of sea bias (i.e. the negative of the actual ocean current components)

n_U, n_V, n_W = U, V, W components of ZMWG fluctuation noise

The error model of Eqn. C22 includes five error states to describe Doppler-based error propagation, and these error states are defined in terms of the following first-order Markov processes:

$$\dot{SF}_U = (-1/\tau_{SF})\, SF_U + uSF_U ;\ uSF_U \text{ ZMWG}$$

$$\dot{B}_V = (-1/\tau_{BV})\, B_V + uB_V ;\ uB_V \text{ ZMWG}$$

$$\dot{B}_W = (-1/\tau_{BW})\, B_W + uB_W ;\ uB_W \text{ ZMWG}$$

$$\dot{SB}_N = (-1/\tau_S)\, SB_N + uSB_N ;\ uSB_N \text{ ZMWG}$$

$$\dot{SB}_E = (-1/\tau_S)\, SB_E + uSB_E ;\ uSB_E \text{ ZMWG} \qquad \text{(C23)}$$

These equations have already appeared as part of the set of 24 scalar differential equations that describe the complete set of Kalman filter error states in Appendix B. Substituting Eqns. C11, C21 and C22 into C20 yields expressions for scalar measurements z_3, z_4 and z_5 in the general observation equation form of Eqn. C5. We identify the following non-zero components of H_3, H_4 and H_5:

$$h_{3,4} = \cos\theta \cos\psi ;\ h_{3,5} = \cos\theta \sin\psi ;\ h_{3,6} = \sin\theta$$

$$h_{3,20} = -U_T ;\ h_{3,23} = -\cos\theta \cos\psi ;\ h_{3,24} = -\cos\theta \sin\psi$$

$$h_{4,4} = \sin\theta \sin\phi \cos\psi - \cos\phi \sin\psi$$

$$h_{4,5} = \sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi ;\ h_{4,6} = -\cos\theta \sin\phi$$

$$h_{4,21} = -U_T ;\ h_{4,23} = \cos\phi \sin\psi - \sin\theta \sin\phi \cos\psi$$

$$h_{4,24} = -\cos\phi \cos\psi - \sin\theta \sin\phi \sin\psi$$

$$h_{5,4} = \sin\theta \cos\phi \cos\psi + \sin\phi \sin\psi$$

$$h_{5,5} = \sin\theta \cos\phi \sin\psi - \sin\phi \cos\psi$$

$$h_{5,6} = -\cos\theta \cos\phi ;\ h_{5,22} = -U_T \qquad \text{(C24)}$$

The discrete measurement noise processes associated with z_3, z_4 and z_5 are identified as $v_3 = -n_U$, $v_4 = -n_V$ and $v_5 = -n_W$ respectively. Thus, each measurement noise process has a noise variance equal to that of the corresponding Doppler channel fluctuation noise. Note that, for a few of the elements of H, it is required to know U_T, the true U body velocity component. A reasonable approximation to U_T can be computed from

$$U_T = U_{DC} / (1 + SF_U) \qquad \text{(C25)}$$

where SF_U would be the latest value of Doppler scale factor error as estimated by the Kalman filter.

APPENDIX D

Measurement Averaging (Prefiltering)

Recall that the update rate for the Kalman filter has been set at 0.1 Hz, whereas the two measurement types have data available at much higher data rates. In the case of the Doppler velocity sensor the rate is 2 Hz, and for the ARNAV Loran-C receiver the data rate is 1 Hz. In order to use all measurement samples during the 10 second Kalman filter update interval, a technique known as measurement averaging, or prefiltering, is employed (see Ref. 8). The use of measurement averaging affects the computation of **z**, **H**, **v** and **R**, as well as the order in which the five Kalman filter recursion equations are executed.

In general, let $\Delta T = t_{k+1} - t_k$ be the fundamental Kalman filter update interval, and assume that a set of vector measurements occurs at N equally spaced time points within ΔT, i.e.

$$\mathbf{z}(t_k + \delta t_i) \; ; \quad i = 1,2,.....,N \tag{D1}$$

where $t_k + \delta t_i$ is the sample time of the i^{th} measurement within the interval ΔT. Recall the basic vector/matrix form of the observation equation:

$$\mathbf{z}(t) = \mathbf{H}(t)\ \delta\mathbf{x}(t) + \mathbf{v}(t) \tag{D2}$$

For the vector measurement at time $t_k + \delta t_i$ we can write

$$\mathbf{z}(t_k + \delta t_i) = \mathbf{H}(t_k + \delta t_i)\ \delta\mathbf{x}(t_k + \delta t_i) + \mathbf{v}(t_k + \delta t_i) \tag{D3}$$

But, in general, we can express

$$\delta\mathbf{x}(t_k + \delta t_i) = \Phi(t_k, t_k + \delta t_i)\ \delta\mathbf{x}(t_k) \tag{D4}$$

where $\Phi(t_k, t_k + \delta t_i)$ is an intermediate value of the transition matrix whose elements can be expressed as follows:

$$\Phi_{i,j}(t_k, t_k + \delta t_i) = \int_{t_k}^{t_k + \delta t_i} F_{i,j}(t)\, dt \; ; \quad i,j = 1,.....,24 \; ; \quad i \neq j$$

$$\Phi_{i,i}(t_k, t_k + \delta t_i) = 1 + \int_{t_k}^{t_k+\delta t_i} F_{i,i}(t)\, dt \; ; \quad i = 1,.....,10$$

$$\Phi_{i,i}(t_k, t_k + \delta t_i) = \exp(-\delta t_i / \tau_i) \; ; \quad i = 11,.....,24 \qquad \text{(D5)}$$

Note that all of the integrals in Eqns. D5 would be computed numerically, using trapezoidal integration, so that the required intermediate values would automatically be available from the trapezoidal integration routines. Substituting Eqn. D4 into Eqn. D3 produces

$$\mathbf{z}(t_k + \delta t_i) = \mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)\ \delta\mathbf{x}(t_k) + \mathbf{v}(t_k + \delta t_i) \qquad \text{(D6)}$$

Summing both sides of Eqn. D6 over all values of $i = 1,...,N$ yields

$$\sum_{i=1}^{N} \mathbf{z}(t_k + \delta t_i) = \sum_{i=1}^{N} [\mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)\ \delta\mathbf{x}(t_k)] + \sum_{i=1}^{N} \mathbf{v}(t_k + \delta t_i)$$

$$= [\sum_{i=1}^{N} \mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)]\ \delta\mathbf{x}(t_k) + \sum_{i=1}^{N} \mathbf{v}(t_k + \delta t_i) \qquad \text{(D7)}$$

Divide both sides of Eqn. D7 by N to get

$$(1/N) \sum_{i=1}^{N} \mathbf{z}(t_k + \delta t_i) = [(1/N) \sum_{i=1}^{N} \mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)]\ \delta\mathbf{x}(t_k) + (1/N) \sum_{i=1}^{N} \mathbf{v}(t_k + \delta t_i) \qquad \text{(D8)}$$

Now define the following set of <u>averaged</u> quantities, all referenced to discrete time t_k:

$$\bar{\mathbf{z}}(k) = (1/N) \sum_{i=1}^{N} \mathbf{z}(t_k + \delta t_i)\text{: averaged measurement}$$

$$\bar{\mathbf{H}}(k) = (1/N) \sum_{i=1}^{N} [\mathbf{H}(t_k + \delta t_i)\ \Phi(t_k, t_k + \delta t_i)]\text{: averaged measurement matrix}$$

$$\bar{\mathbf{v}}(k) = (1/N) \sum_{i=1}^{N} \mathbf{v}(t_k + \delta t_i): \text{ averaged measurement noise}$$

$$\bar{\mathbf{R}}(k) = (1/N^2) \sum_{i=1}^{N} \mathbf{R}(t_k + \delta t_i): \text{ averaged measurement noise covariance matrix}$$

$$= (1/N)\ \mathbf{R}\text{; if } \mathbf{R} \text{ is a constant matrix}$$

With the foregoing definitions, Eqn. D8 can be written as

$$\bar{\mathbf{z}}(k) = \bar{\mathbf{H}}(k)\ \delta\mathbf{x}(k) + \bar{\mathbf{v}}(k) \qquad \text{(D9)}$$

and becomes the equivalent <u>averaged</u> observation equation, as referenced to discrete time t_k.

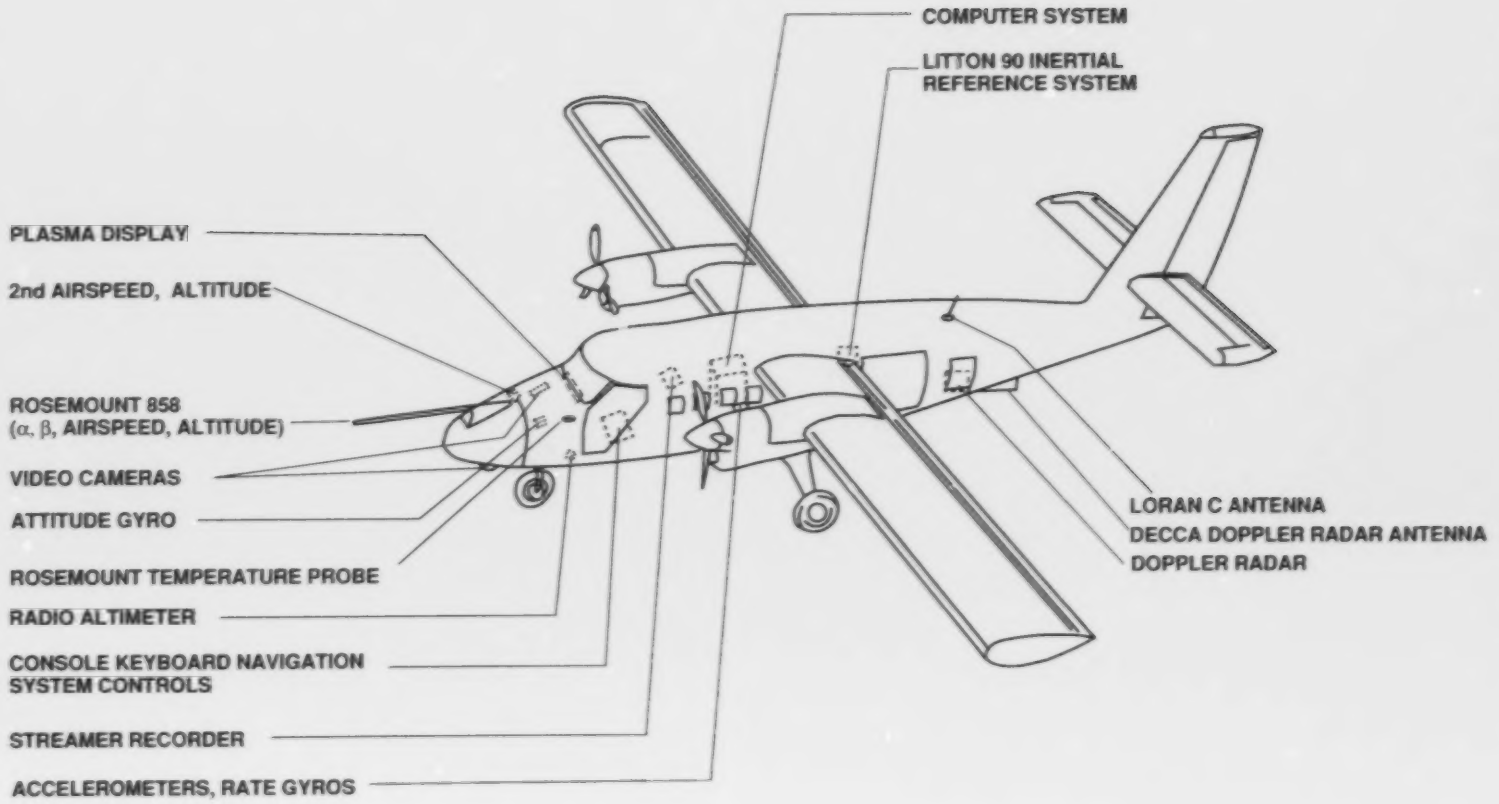

FIG. 1: IAR TWIN OTTER ATMOSPHERIC RESEARCH AIRCRAFT AS INSTRUMENTED FOR AIRBORNE WIND MEASUREMENT

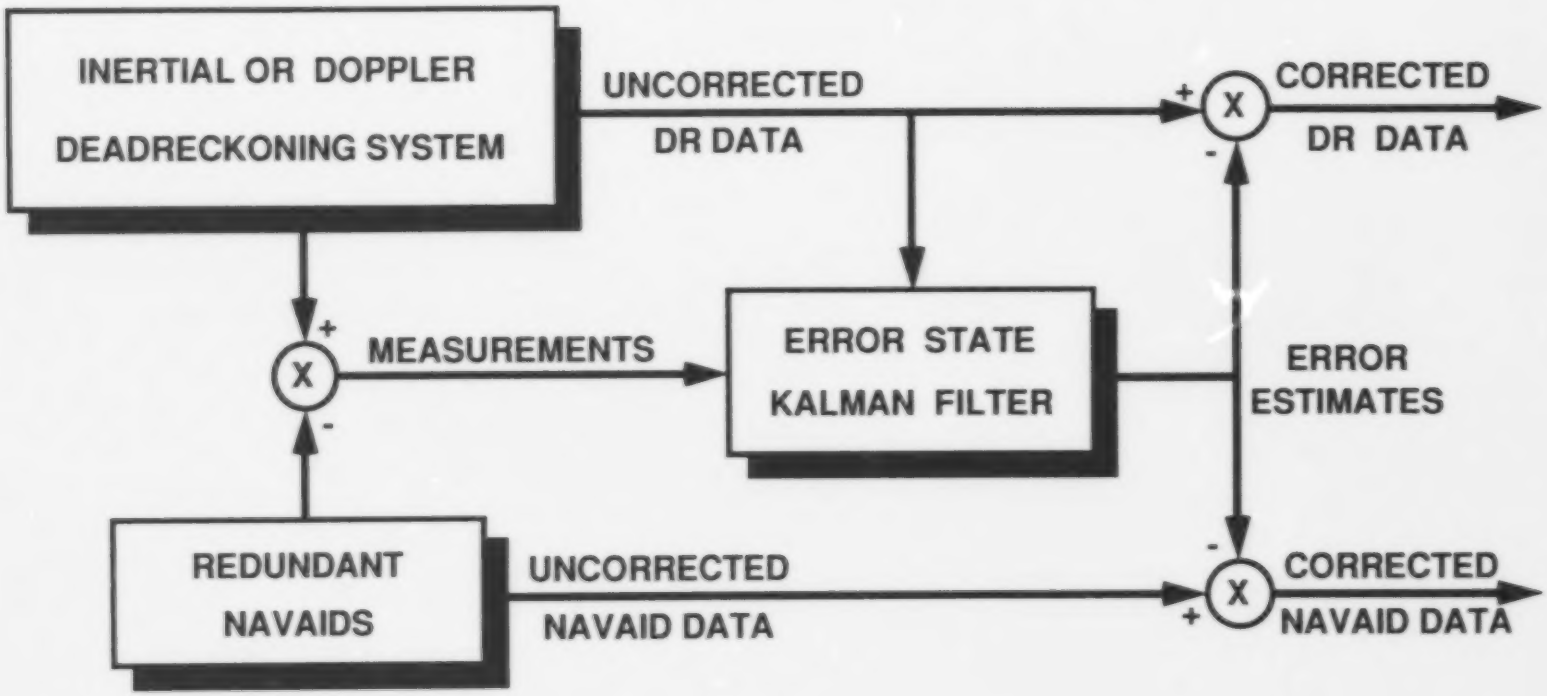

FIG. 2: ERROR STATE FEEDFORWARD KALMAN FILTER NAVIGATOR

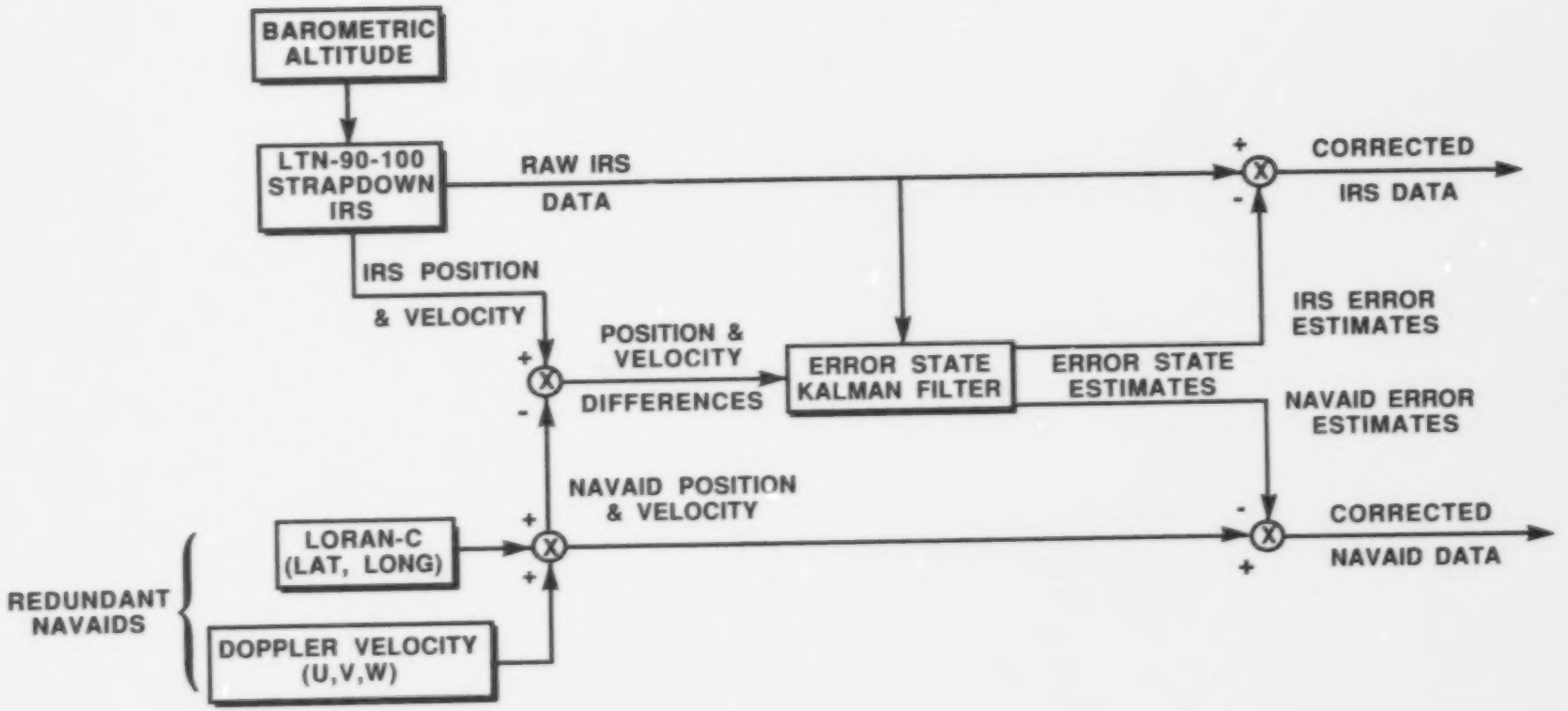

FIG. 3: IRS / DOPPLER / LORAN-C KALMAN FILTER CONFIGURATION

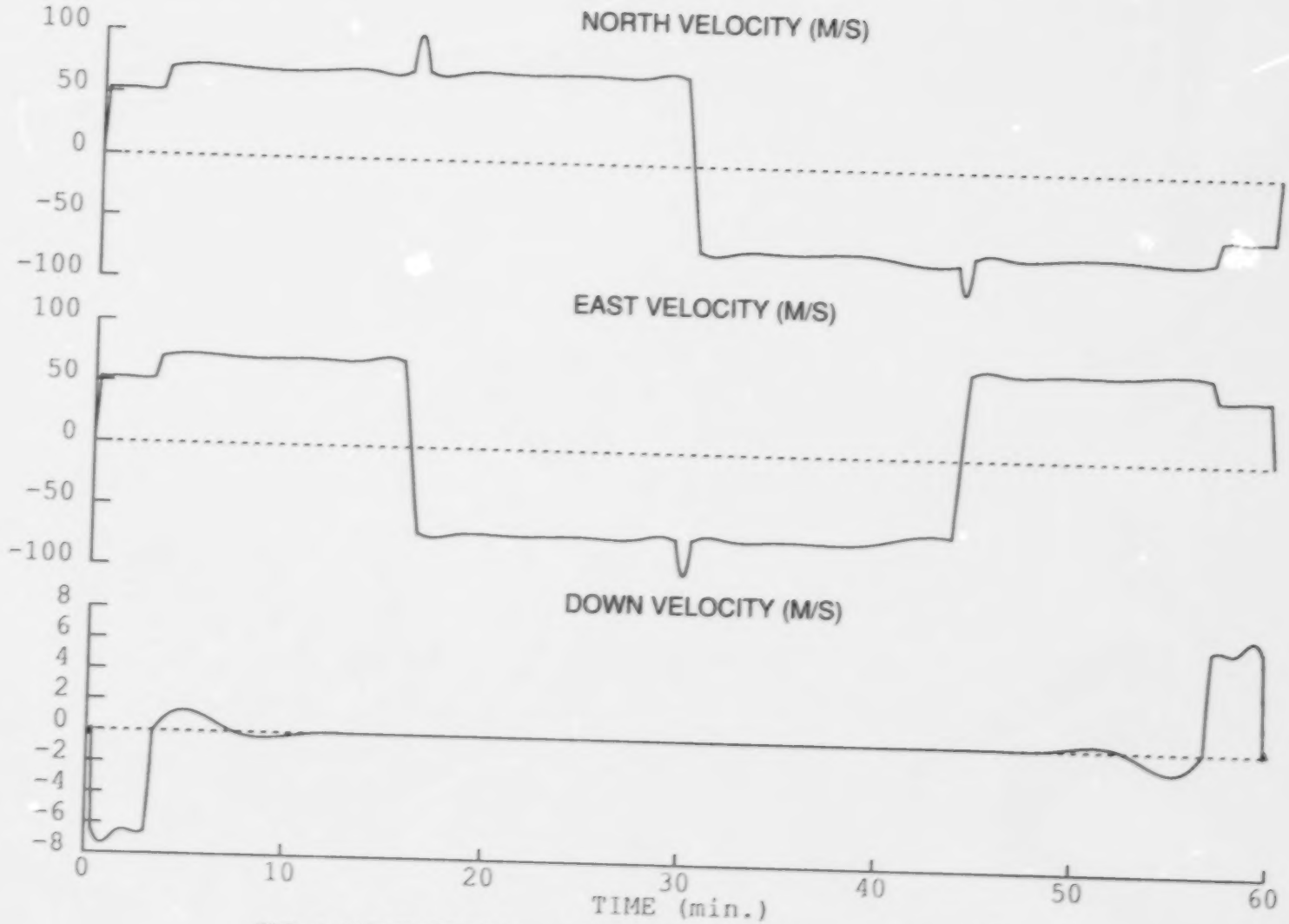

FIG. 4: VELOCITY COMPONENTS OF SIMULATED TRAJECTORY

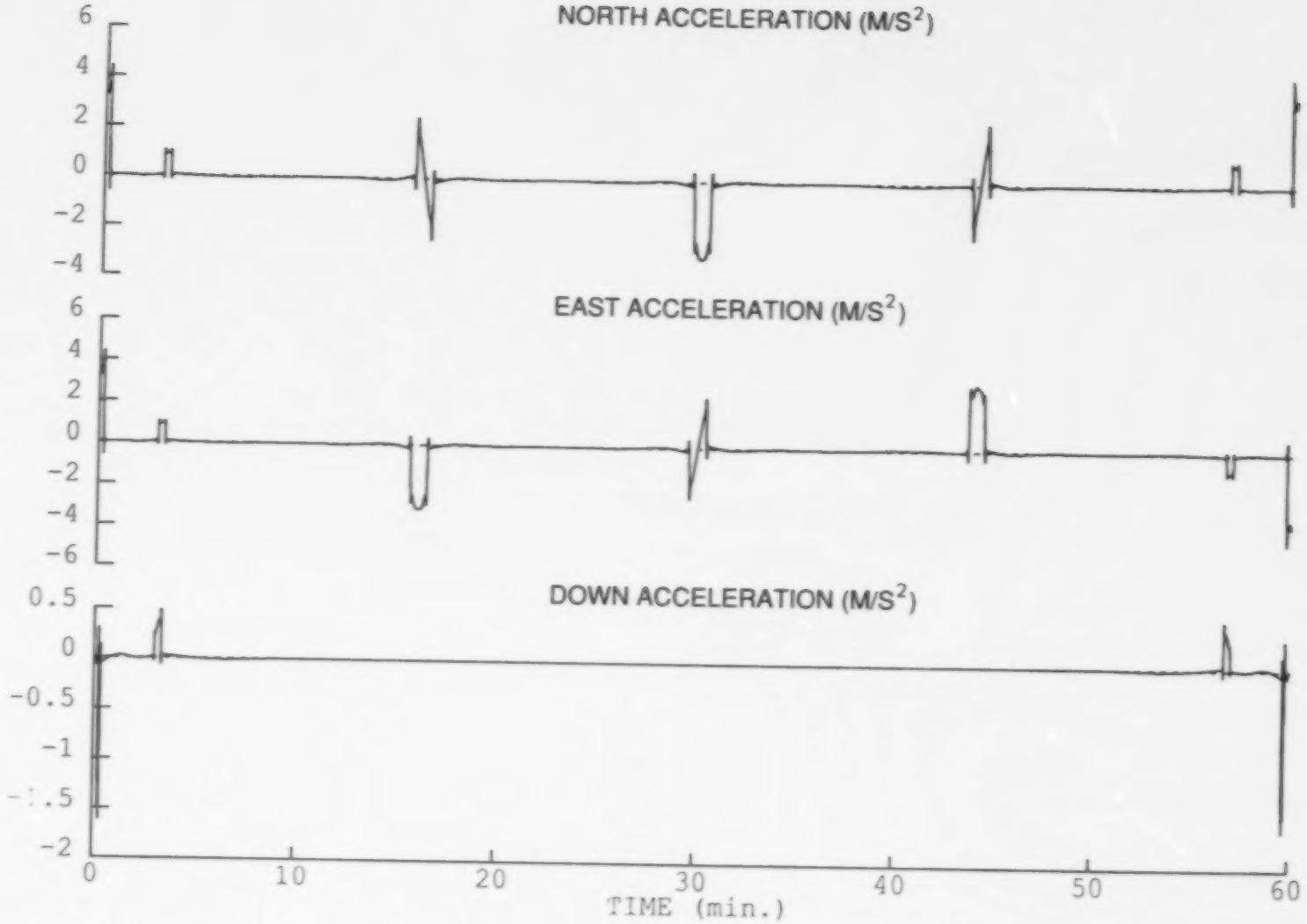

FIG. 5: VELOCITY RATE COMPONENTS OF SIMULATED TRAJECTORY

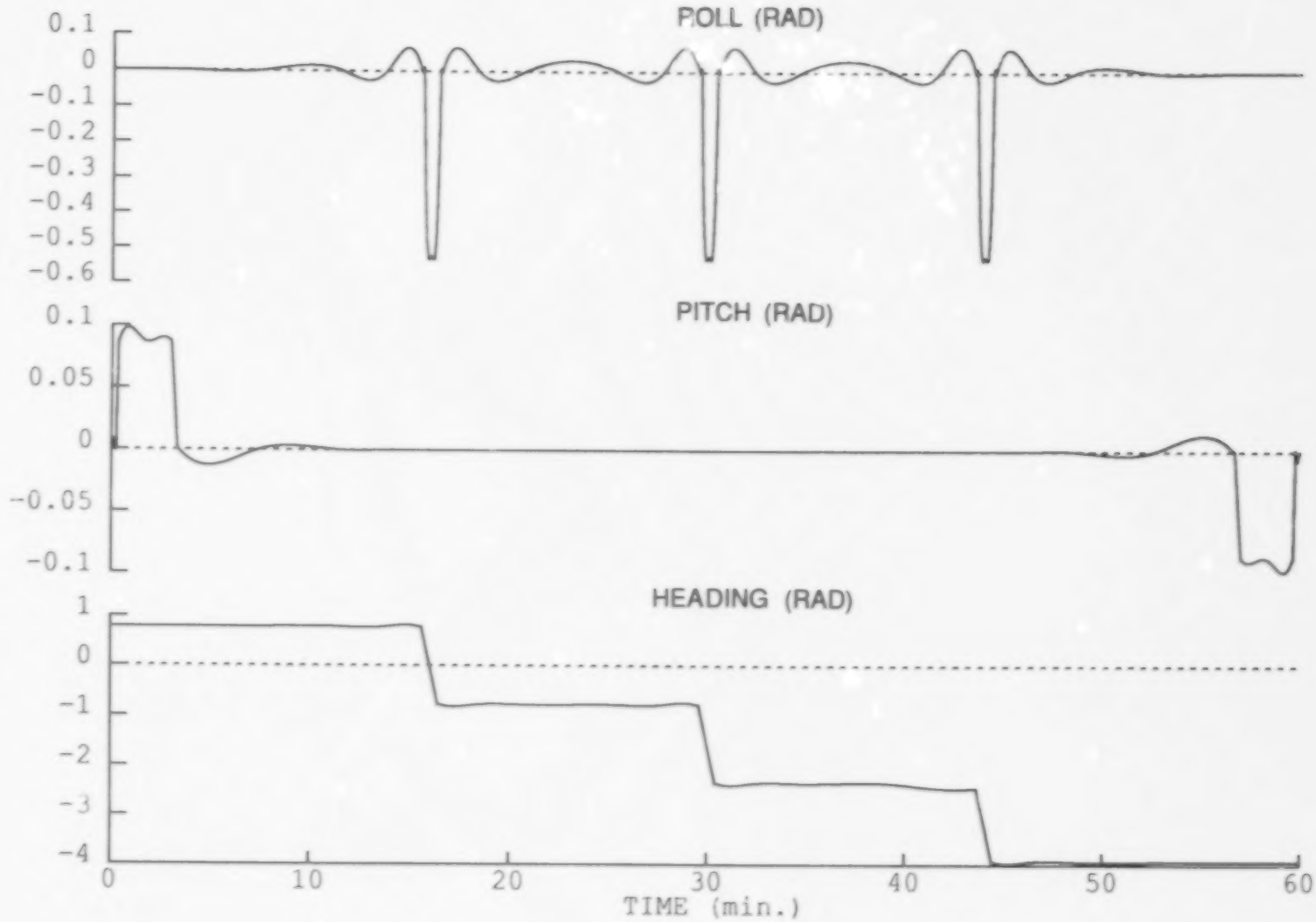

FIG. 6: ATTITUDE COMPONENTS OF SIMULATED TRAJECTORY

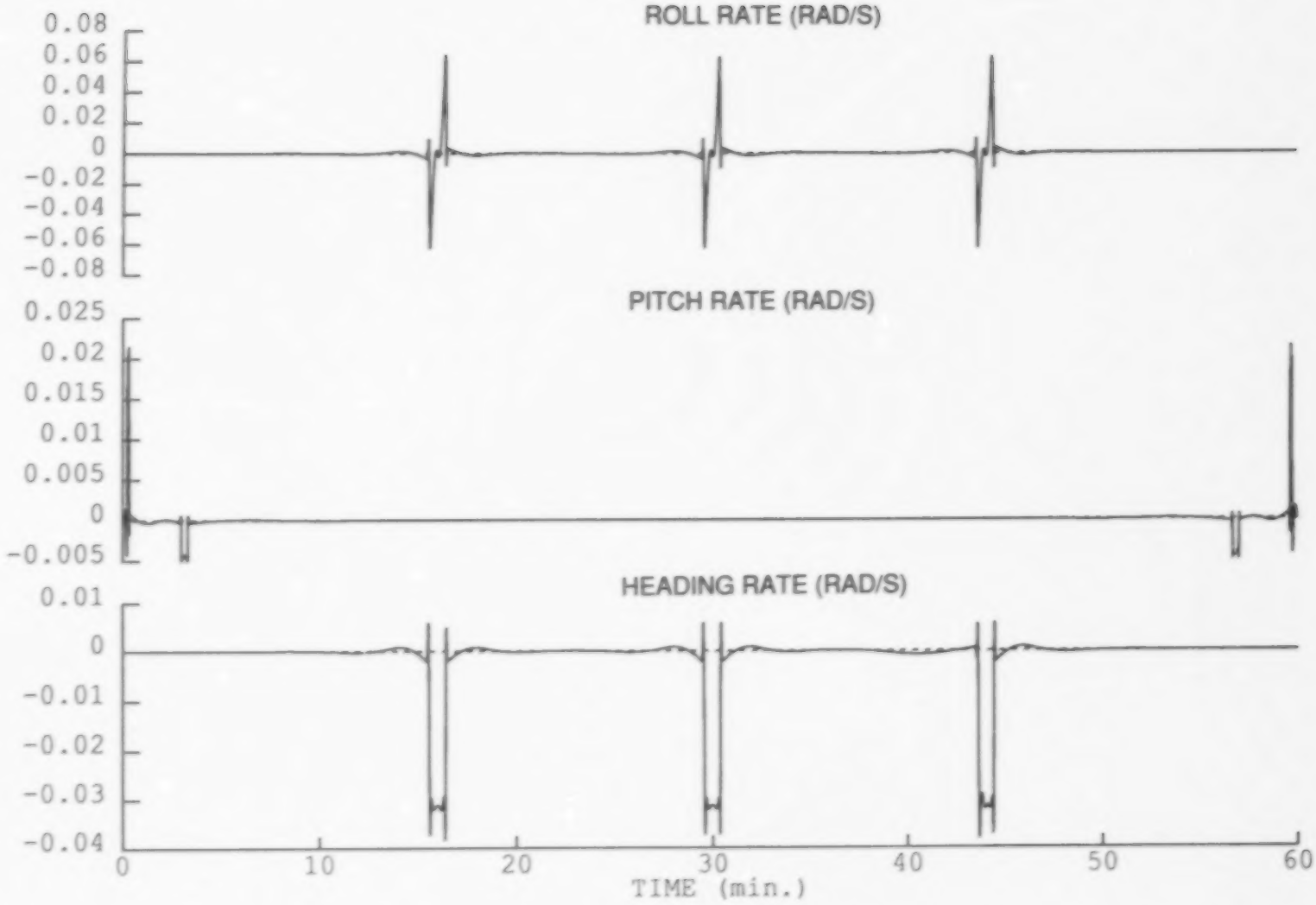

FIG. 7: ATTITUDE RATE COMPONENTS OF SIMULATED TRAJECTORY

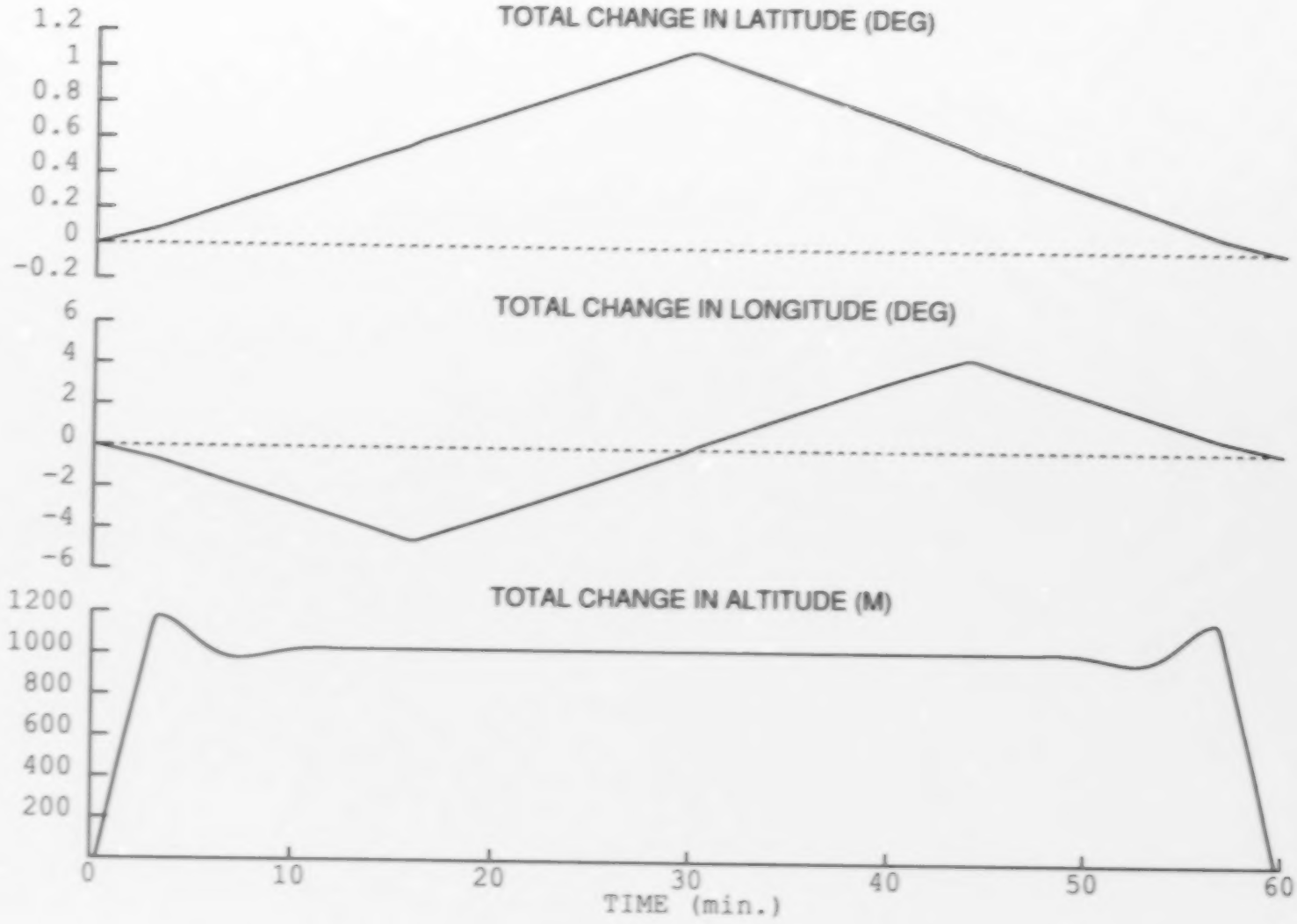

FIG. 8: POSITION CHANGE COMPONENTS OF SIMULATED TRAJECTORY

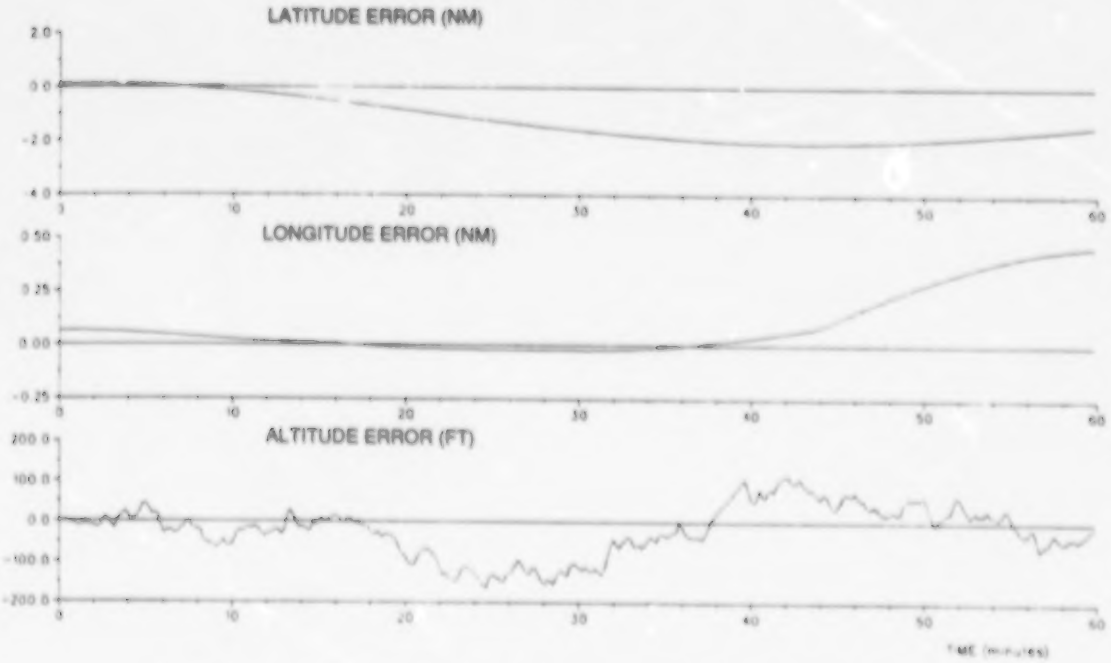

FIG. 9: STRAPDOWN NAVIGATOR POSITION ERRORS

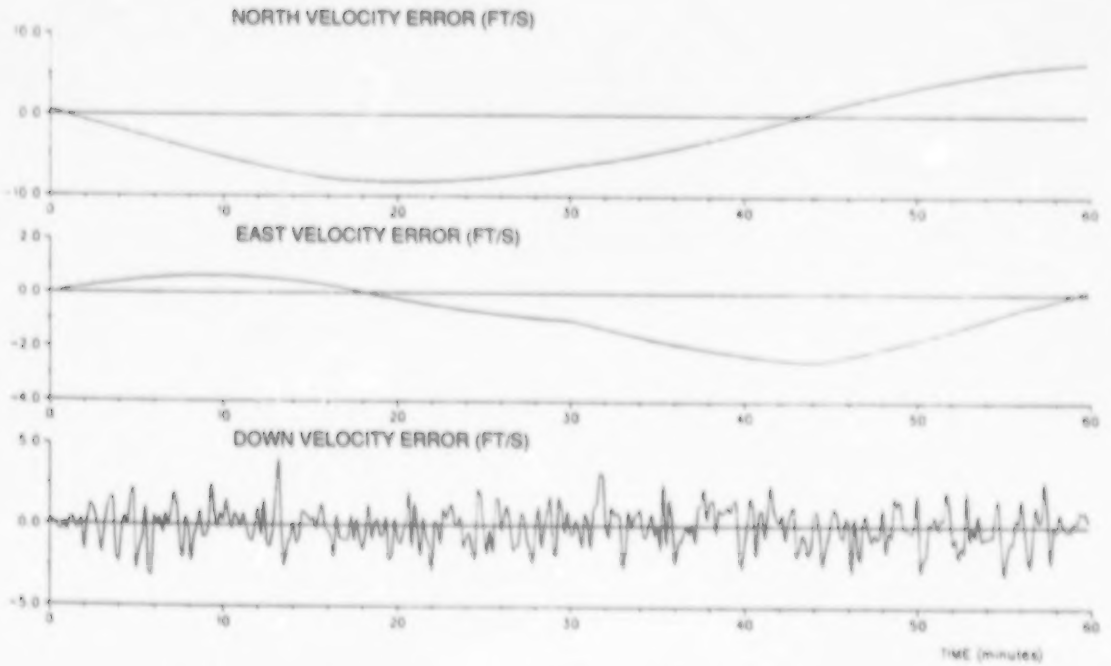

FIG. 10: STRAPDOWN NAVIGATOR VELOCITY ERRORS

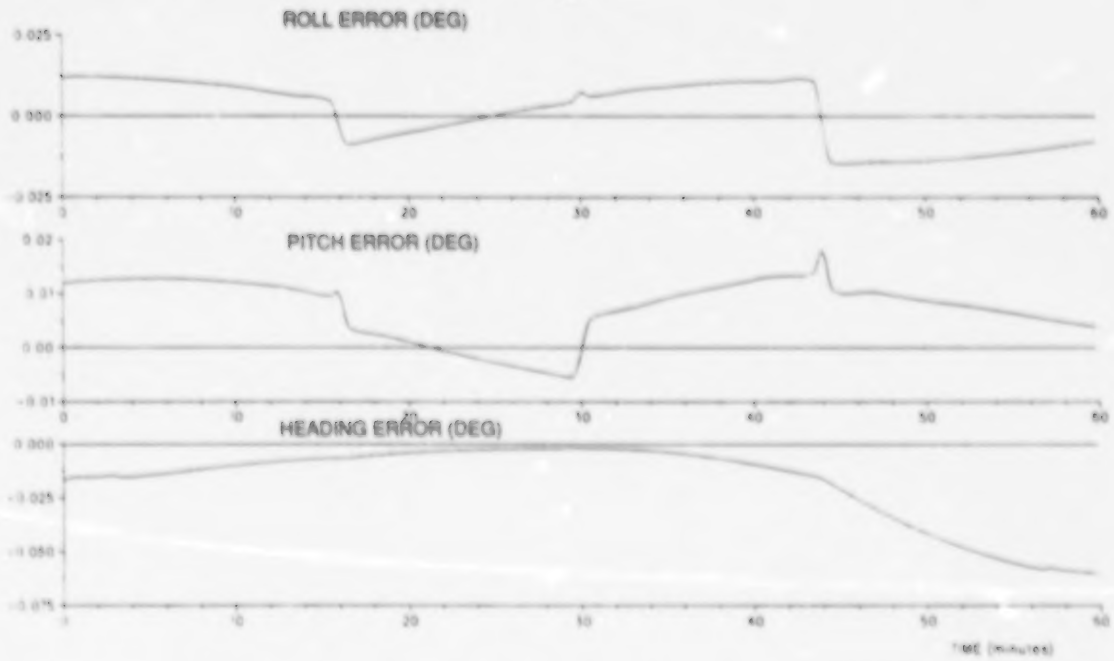

FIG. 11: STRAPDOWN NAVIGATOR ATTITUDE ERRORS

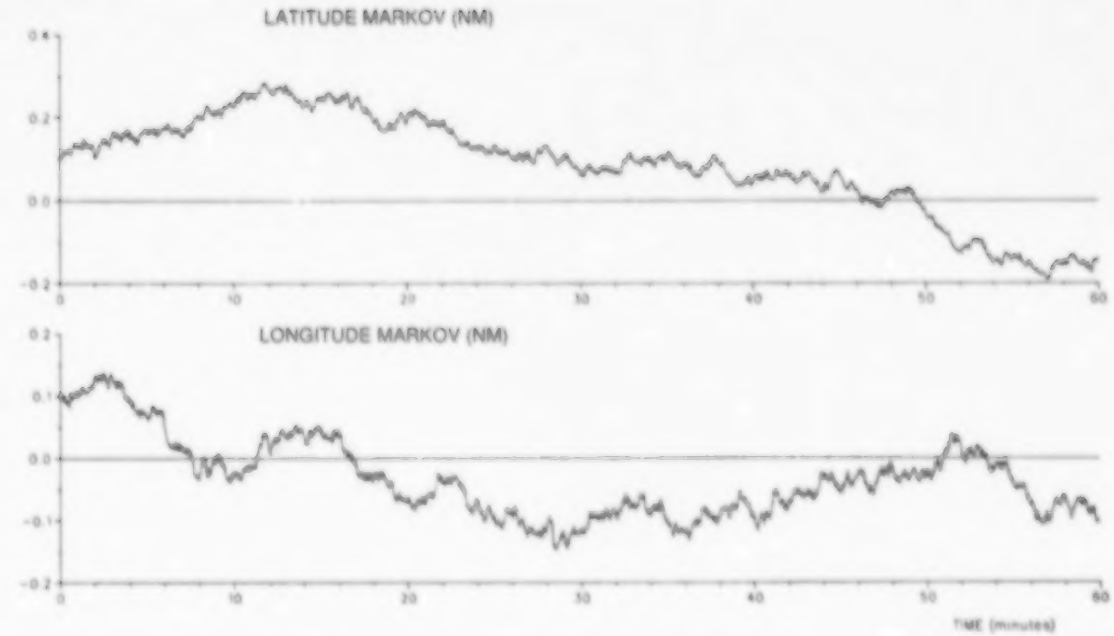

FIG. 12: LORAN-C LATITUDE AND LONGITUDE MARKOV PROCESSES

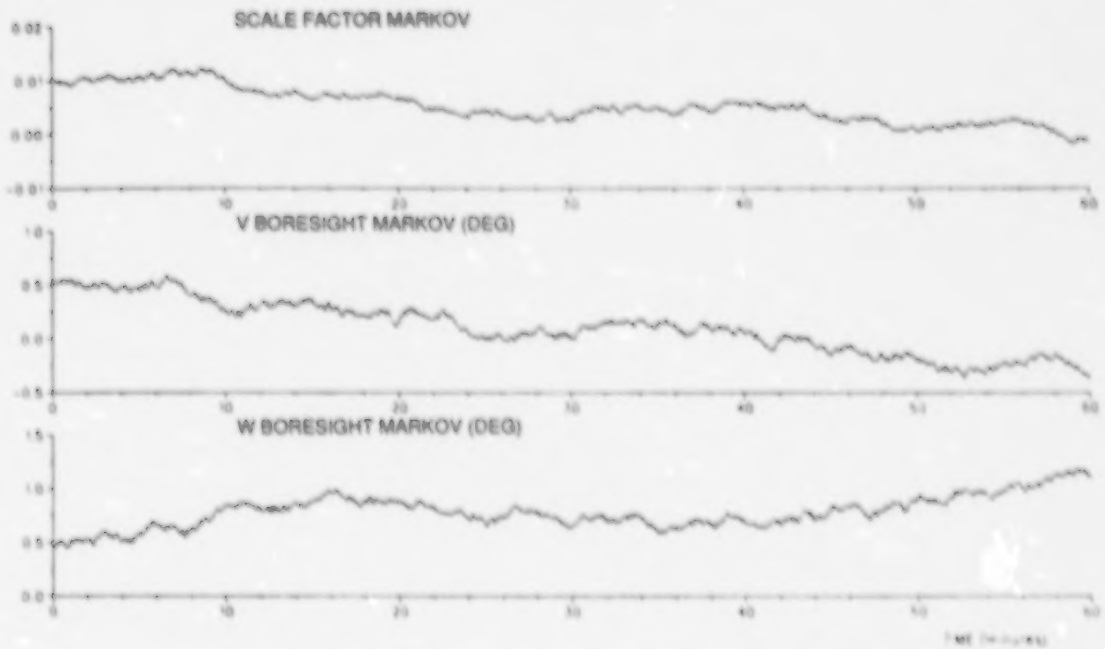

FIG. 13: DOPPLER VELOCITY MARKOV PROCESSES

```
0000                - NO. OF SECONDS OF INITIAL DATA TO BE SKIPPED
0350                - DURATION OF KF RUN (IN SECONDS, DIVISIBLE BY 10)
00000               - START TIME FOR OVERWATER FLIGHT (IN SECS RELATIVE)
00000               - END TIME FOR OVERWATER FLIGHT   (TO START OF FILE)
82.500              - NOMINAL OPERATING LATITUDE (DEGS)
015.97 000.91 -01.54 - U,V,W DOPPLER-TO-INS LEVER ARMS (FEET)
* INS SYSTEM ERROR STATE STATISTICS (1-SIGMA):
1.  0.1000E00 2.  0.1000E00  LAT & LONG (NM)
3.  3.0000E01 4.  0.3300E01  ALT (FT) & N. VELOC. (FT/SEC)
5.  0.3300E01 6.  0.3300E01  EAST & VERT. VELOC'S (FT/SEC)
7.  0.8300E-2 8.  0.8300E-2  N. & E. AXIS ATTITUDE (DEG)
9.  0.8500E-1 10. 0.2500E-1  Z AXIS ATT. (DEG) & BARO LOOP ACC. (FT/S**2)
* SENSOR MARKOV ERROR STATISTICS (1-SIGMA VALUE & CORR. TIME IN SECS.):
11. 2.0000E-2  07500.0       X AXIS GYRO (DEG/HR)
12. 2.0000E-2  07500.0       Y AXIS GYRO (DEG/HR)
13. 2.0000E-2  07500.0       Z AXIS GYRO (DEG/HR)
14. 2.0000E-3  15000.0       X AXIS ACCEL (FT/SEC**2)
15. 2.0000E-3  15000.0       Y AXIS ACCEL (FT/SEC**2)
16. 4.0000E-3  15000.0       Z AXIS ACCEL (FT/SEC**2)
17. 2.0000E02  03000.0       ALTIMETER BIAS (FEET)
18. 0.2000E-0  07200.0       LATITUDE LORAN-C FIX (NM)
19. 0.2000E-0  07200.0       LONGITUDE LORAN-C FIX (NM)
20. 0.0200E00  15000.0       DOPPLER SCALE FACTOR
21. 1.1500E00  15000.0       DOPPLER V BORESIGHT (DEG)
22. 1.1500E00  15000.0       DOPPLER H BORESIGHT (DEG)
23. 4.5000E00  00900.0       NORTH SEA BIAS (FT/SEC)
24. 4.5000E00  00900.0       EAST SEA BIAS (FT/SEC)
* INS SYSTEM PLANT NOISE STATISTICS (1-SIGMA):
1.  2.3000E-6 2.  2.3000E-6  INTEGRATED X & Y GYRO (DEG)
3.  2.3000E-6 4.  1.0000E-3  INTEGRATED Z GYRO (DEG) & X ACC. (FT/SEC)
5.  1.0000E-3 6.  1.0000E-3  INTEGRATED Y & Z ACCEL. (FT/SEC)
7.  1.0000E-3 8.  1.0000E-3  INTEGRATED N & E GRAVITY (FT/SEC)
9.  1.0000E-3 10. 3.1620E01  INTEG'D Z GRAVITY (FT/S) & ALT NOISE (FT-S)
* MEASUREMENT NOISE STATISTICS (UNAVERAGED, 1-SIGMA):
1.  0.0500E00 2.  0.0500E00  LAT & LONG LORAN-C (NM)
3.  5.3400E00 4.  5.3400E00 5.  5.3400E00  U,V,W DOPPLER (FT/SEC)
6.  5.0000E00               BAROMETRIC ALTIMETER (FEET)
```

FIG. 14: LISTING OF FILE SDKFNAV CTRL

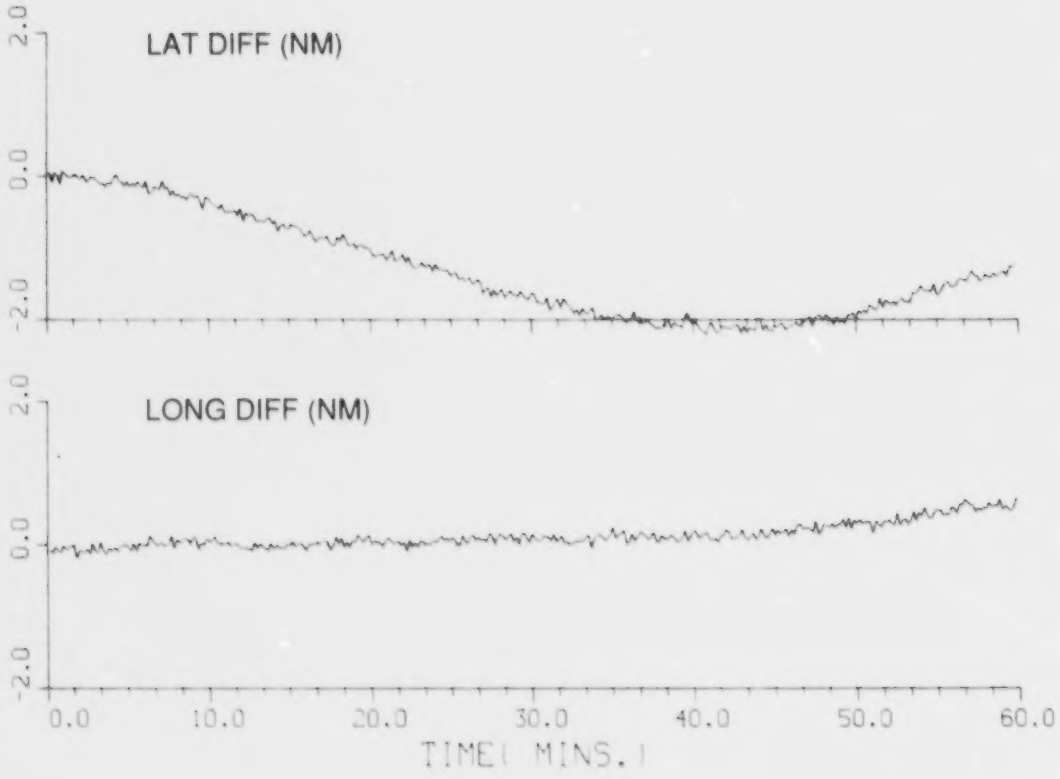

FIG. 15: KALMAN FILTER LORAN-C POSITION MEASUREMENTS

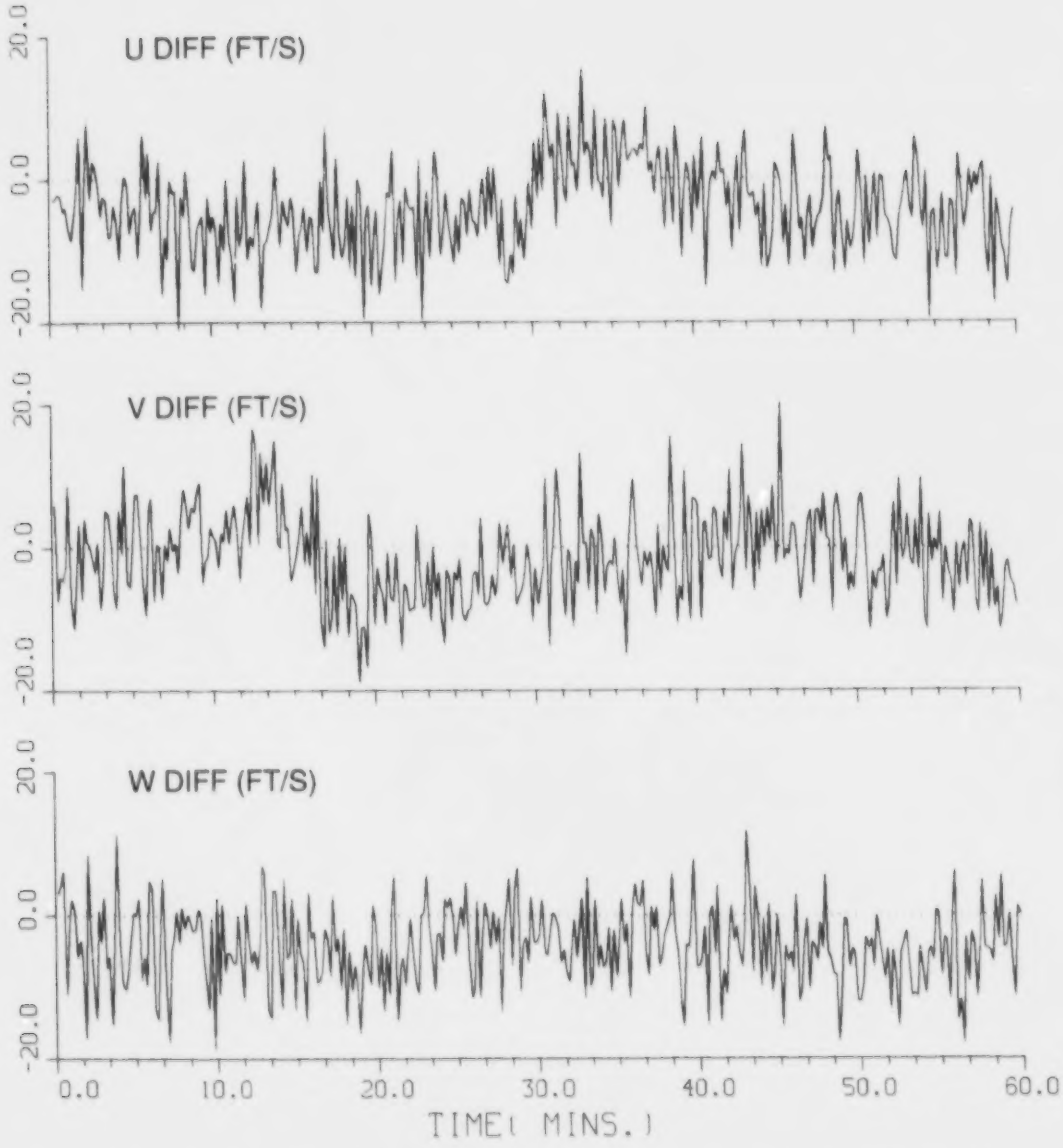

FIG. 16: KALMAN FILTER DOPPLER VELOCITY MEASUREMENTS

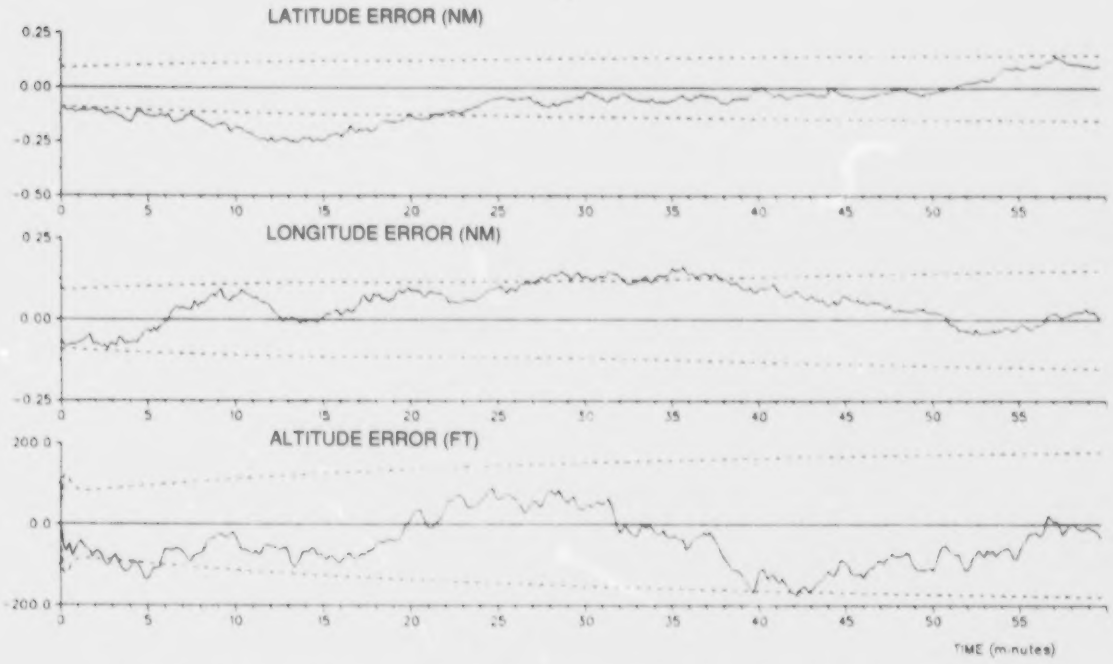

FIG. 17: ACCURACY OF IRS POSITION ERROR STATE ESTIMATION

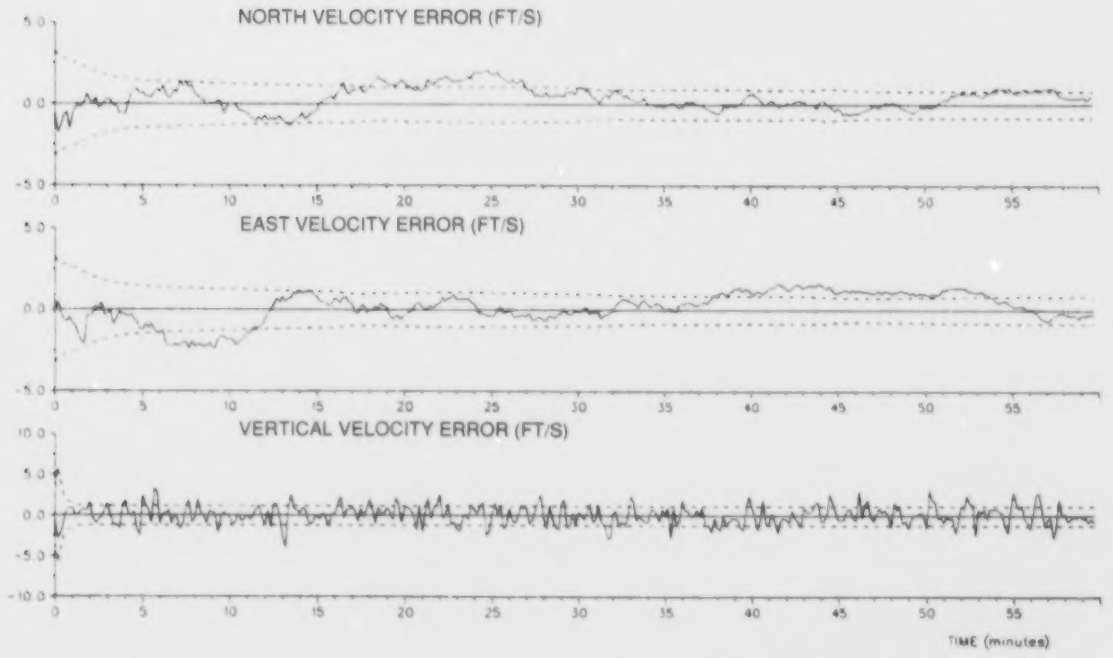

FIG. 18: ACCURACY OF IRS VELOCITY ERROR STATE ESTIMATION

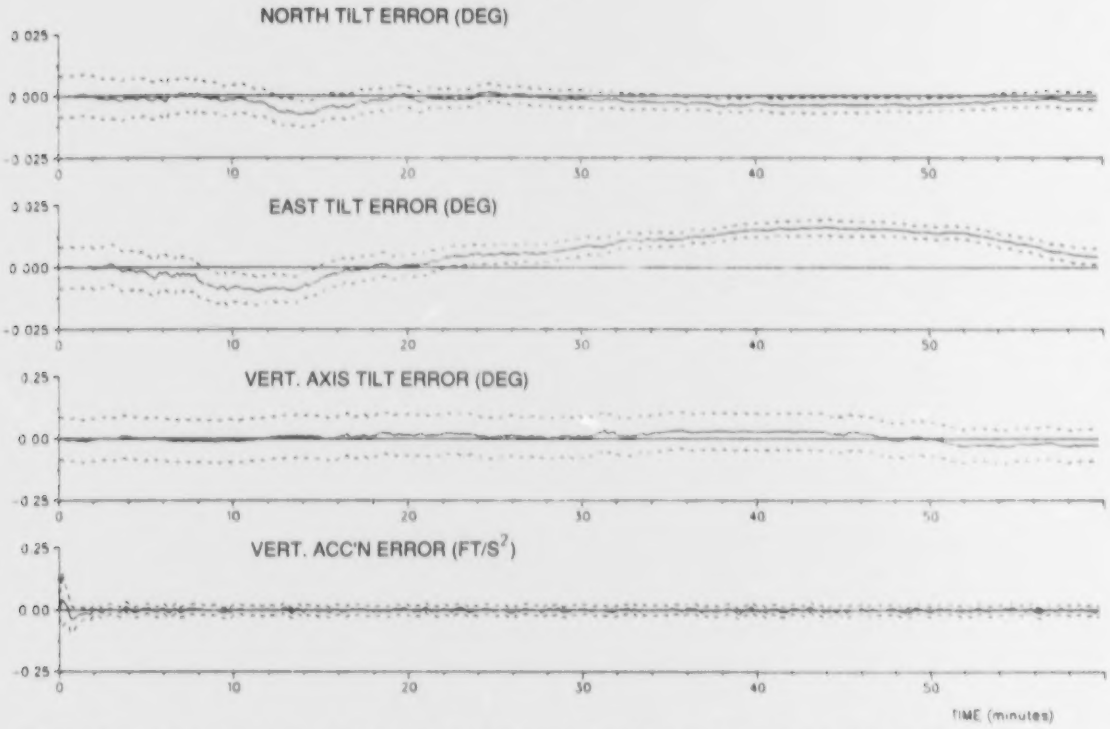

FIG. 19: IRS TILT AND VERTICAL ACCELERATION ERROR STATE ESTIMATES

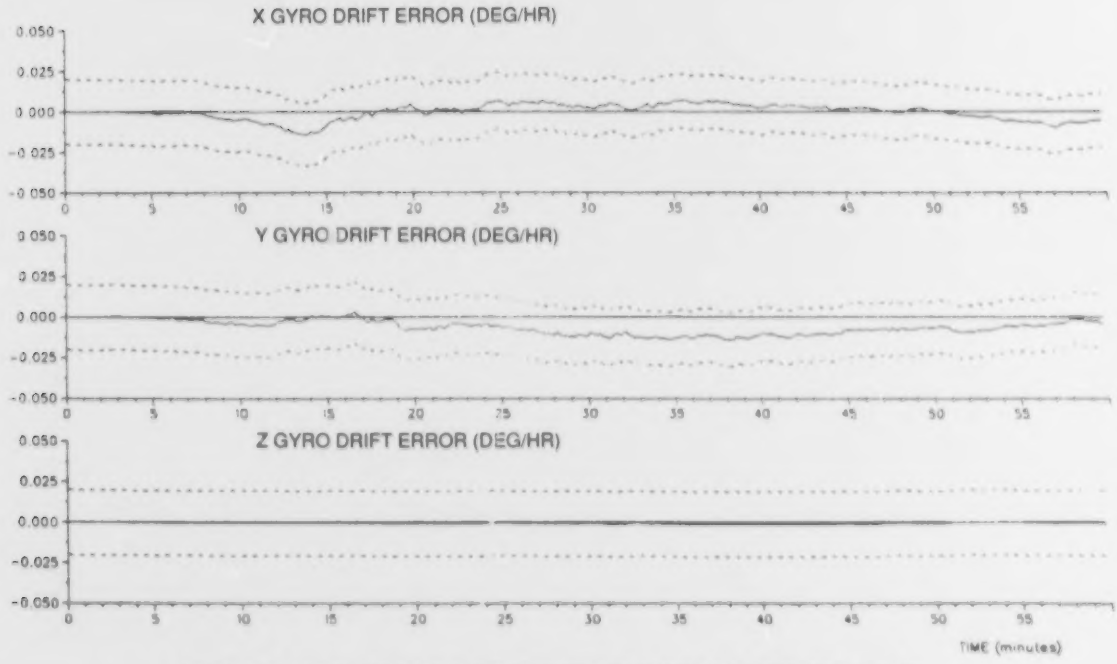

FIG. 20: IRS GYRO DRIFT ERROR STATE ESTIMATES

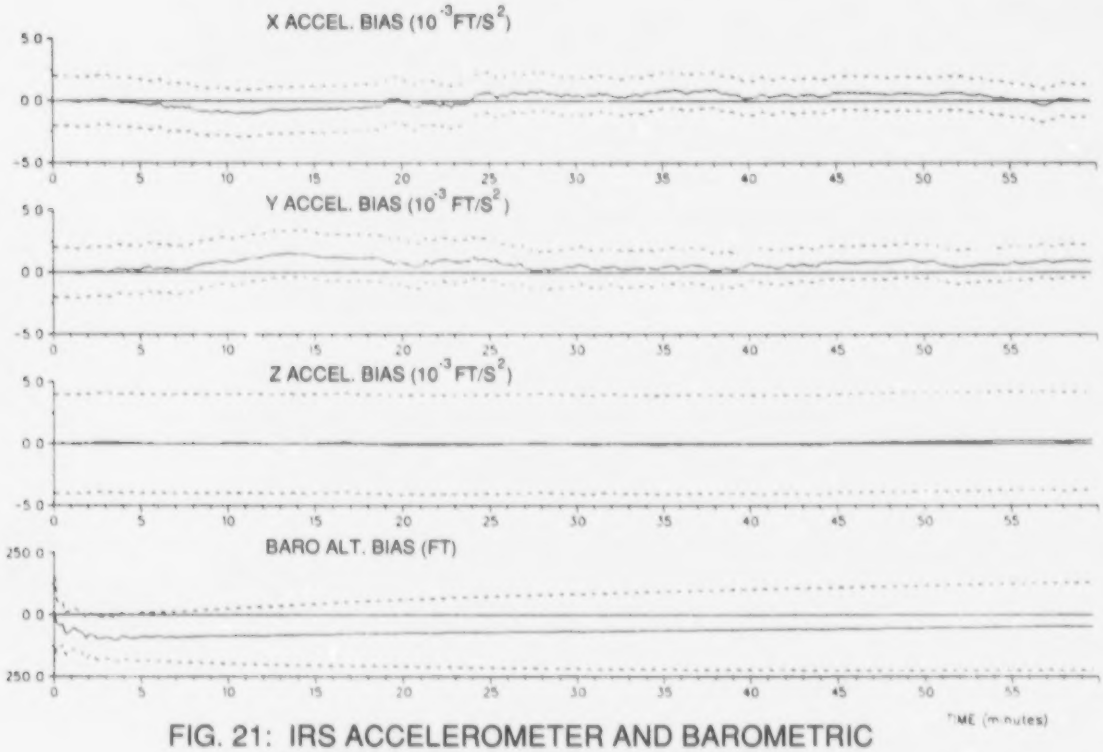

FIG. 21: IRS ACCELEROMETER AND BAROMETRIC ALTIMETER ERROR STATE ESTIMATES

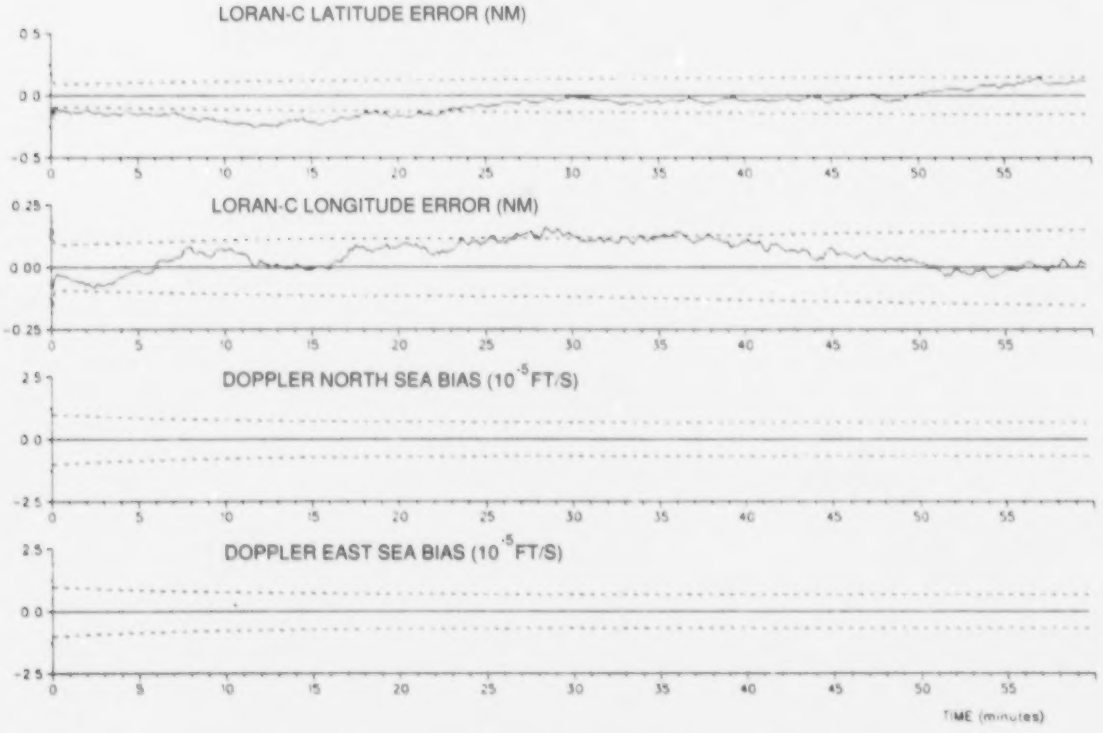
LORAN-C LATITUDE ERROR (NM)
LORAN-C LONGITUDE ERROR (NM)
DOPPLER NORTH SEA BIAS (10^{-5} FT/S)
DOPPLER EAST SEA BIAS (10^{-5} FT/S)
TIME (minutes)

FIG. 22: ACCURACY OF LORAN-C POSITION AND DOPPLER SEA BIAS ERROR STATE ESTIMATION

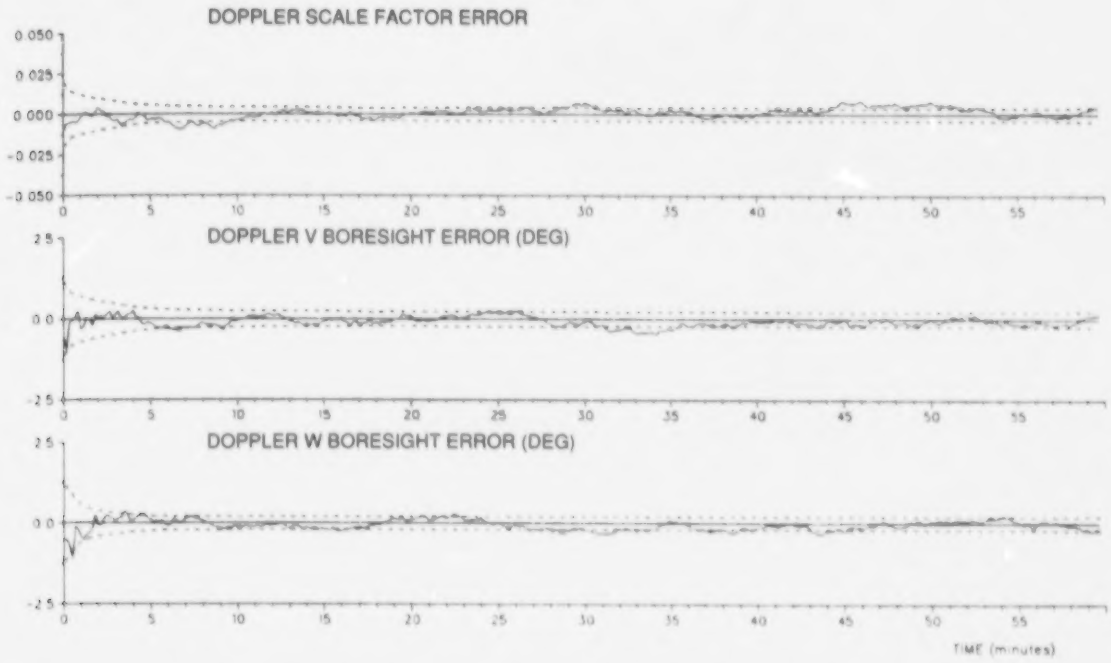

FIG. 23: ACCURACY OF DOPPLER VELOCITY ERROR STATE ESTIMATION

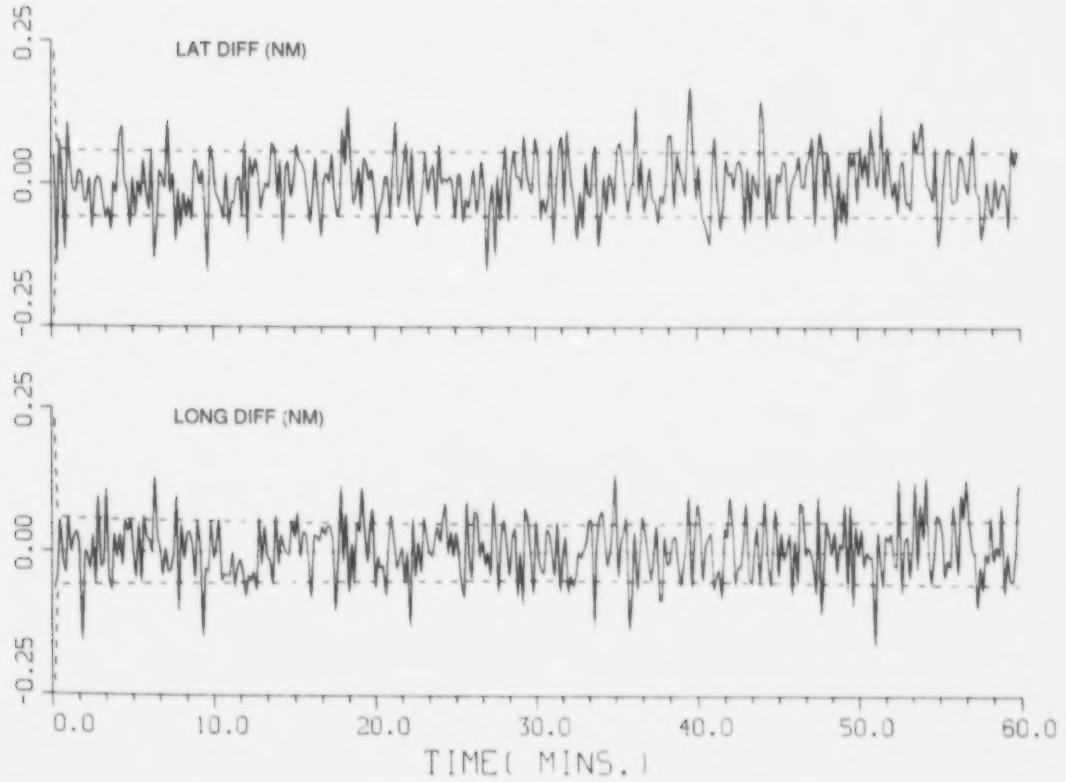

FIG. 24: INNOVATIONS FOR KALMAN FILTER POSITION MEASUREMENTS

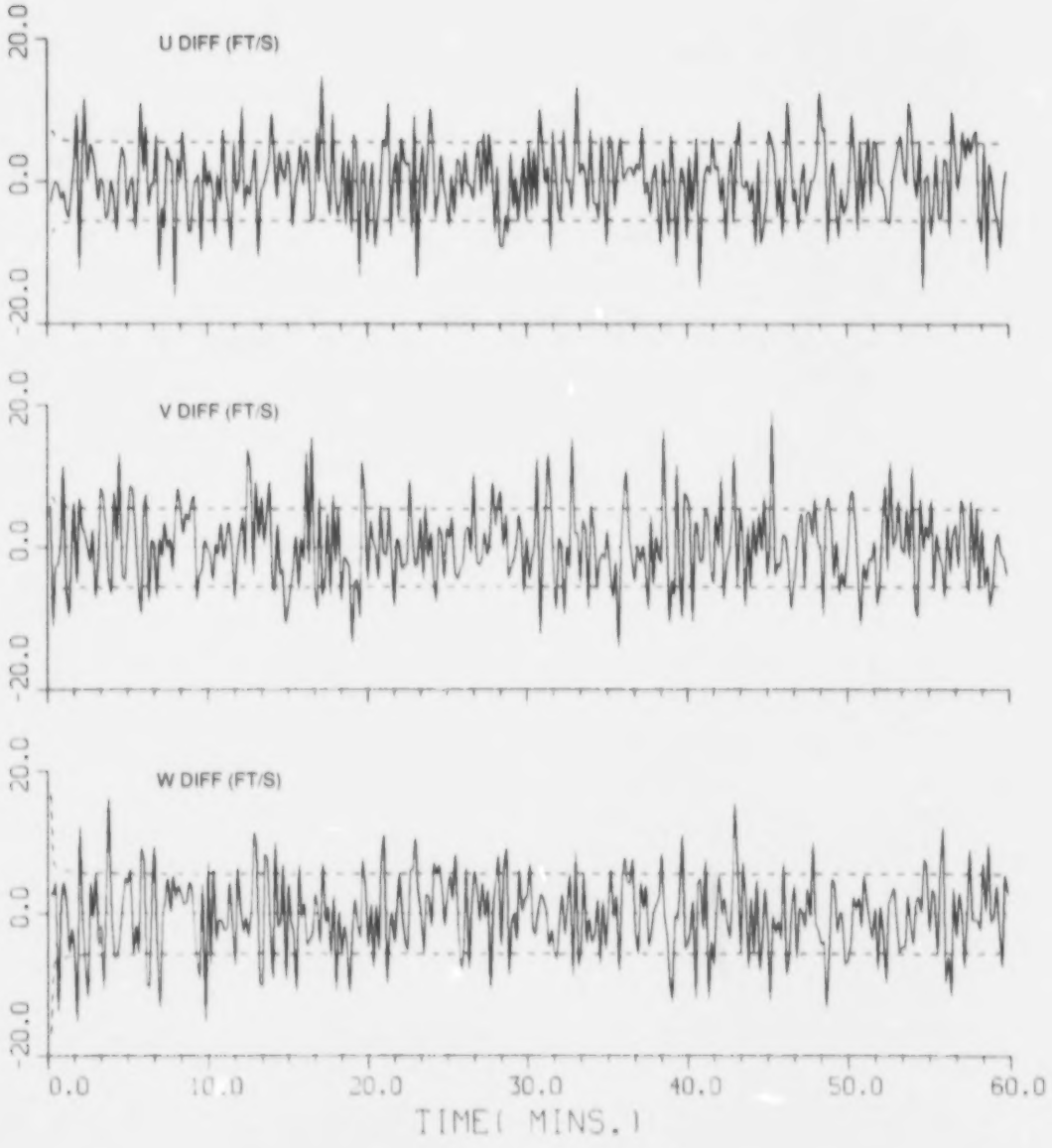

FIG. 25: INNOVATIONS FOR KALMAN FILTER VELOCITY MEASUREMENTS

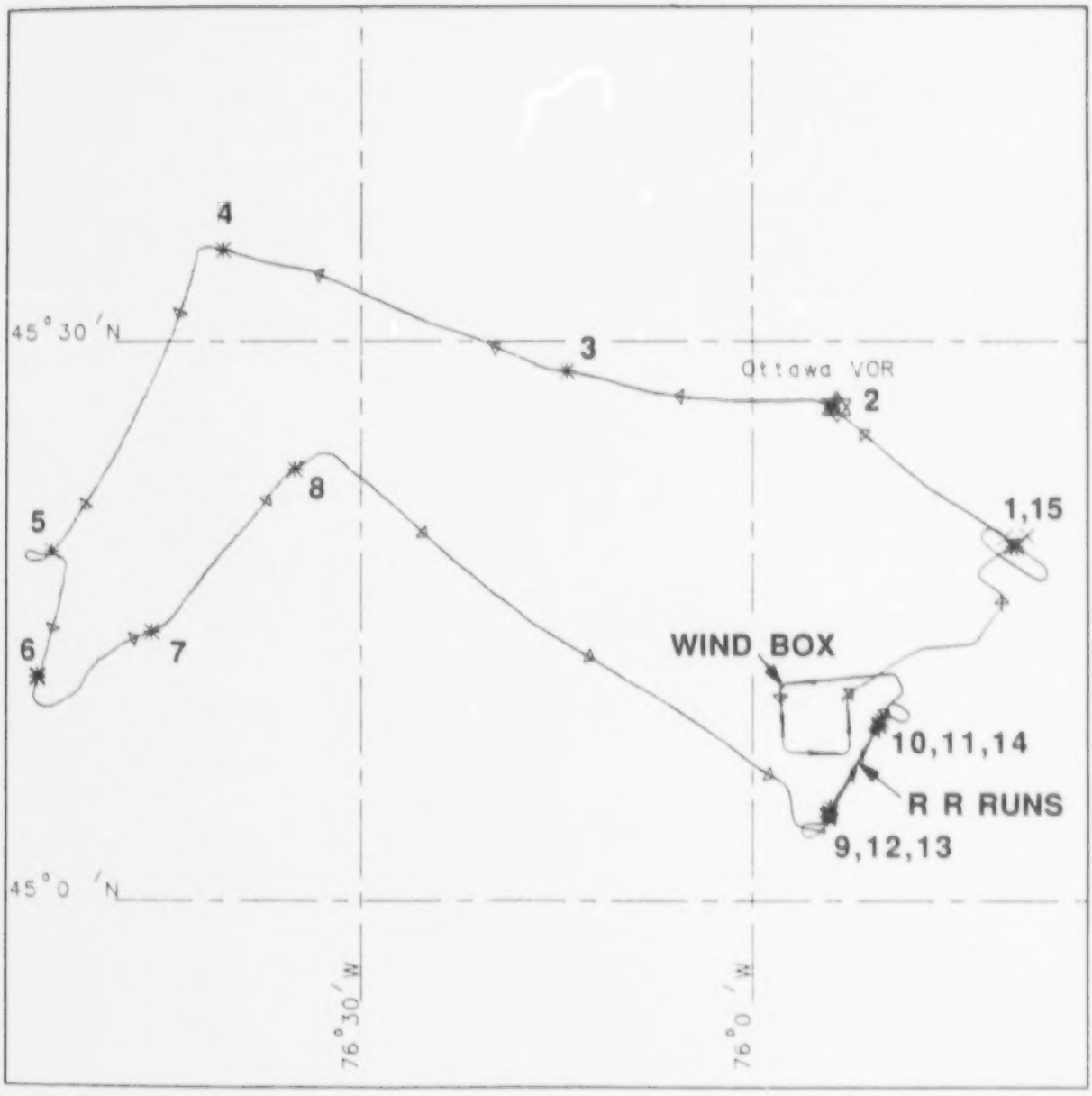

FIG. 26: HORIZONTAL TRACK PLOT FOR FLIGHT #2

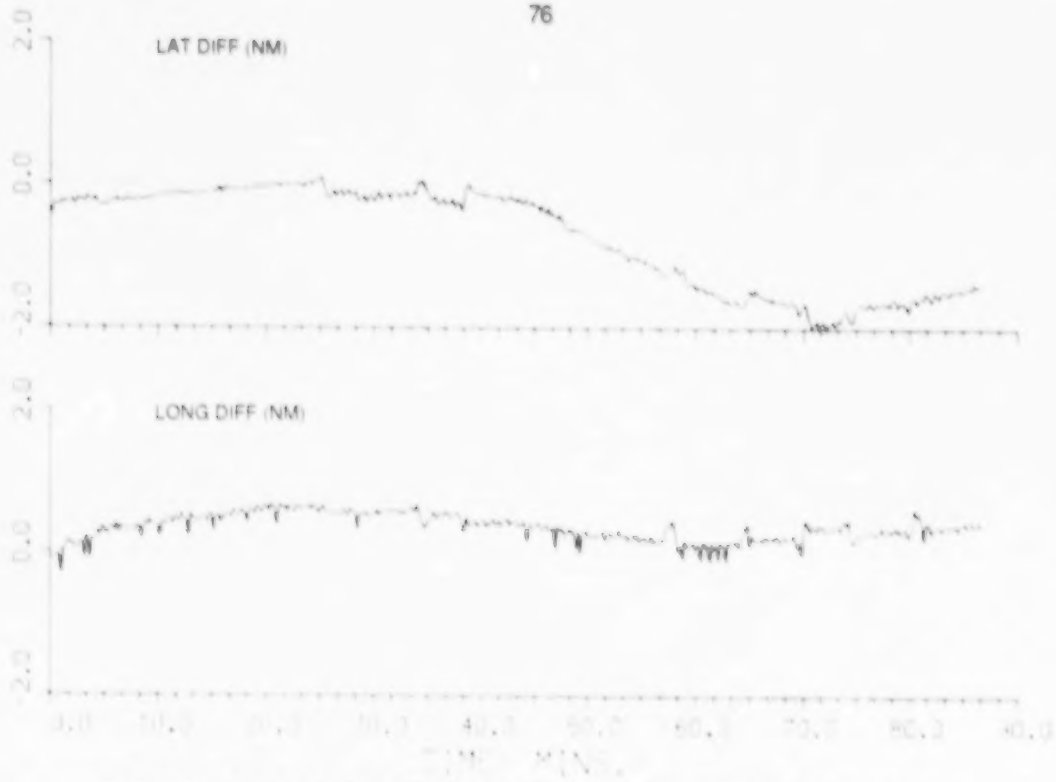

FIG. 27: LORAN -C POSITION MEASUREMENTS FOR FLIGHT #1
LORAN-C DATA PROBLEMS

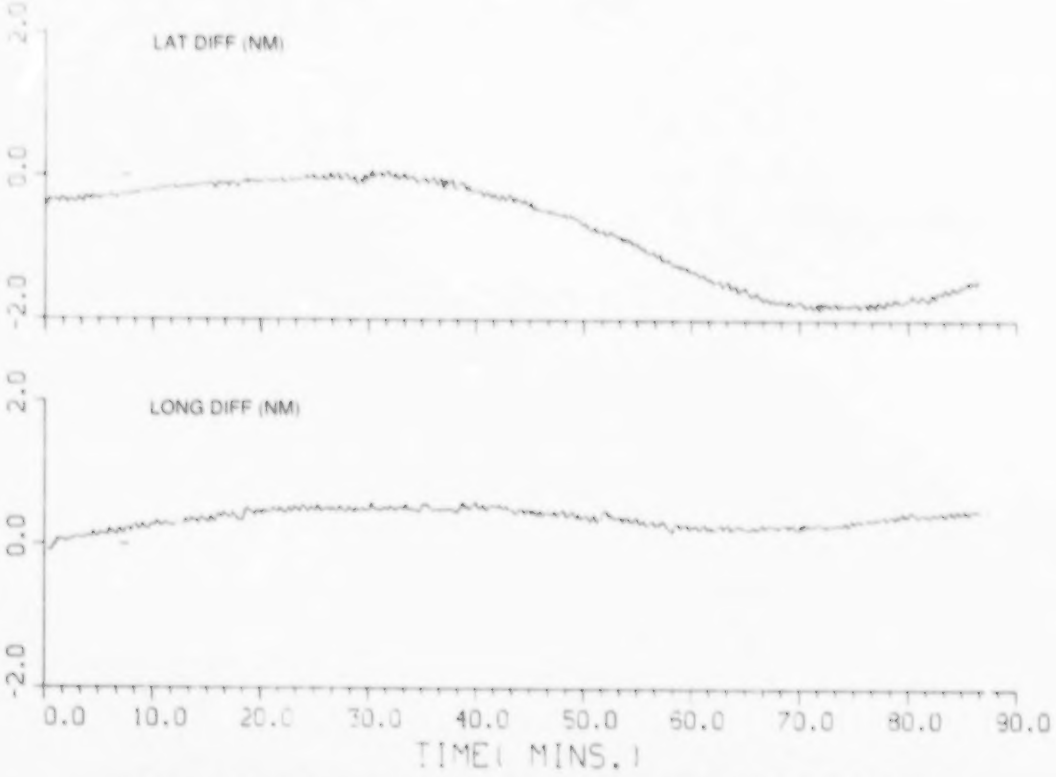

FIG. 28: LORAN-C POSITION MEASUREMENTS FOR FLIGHT #1
PROBLEMS CORRECTED

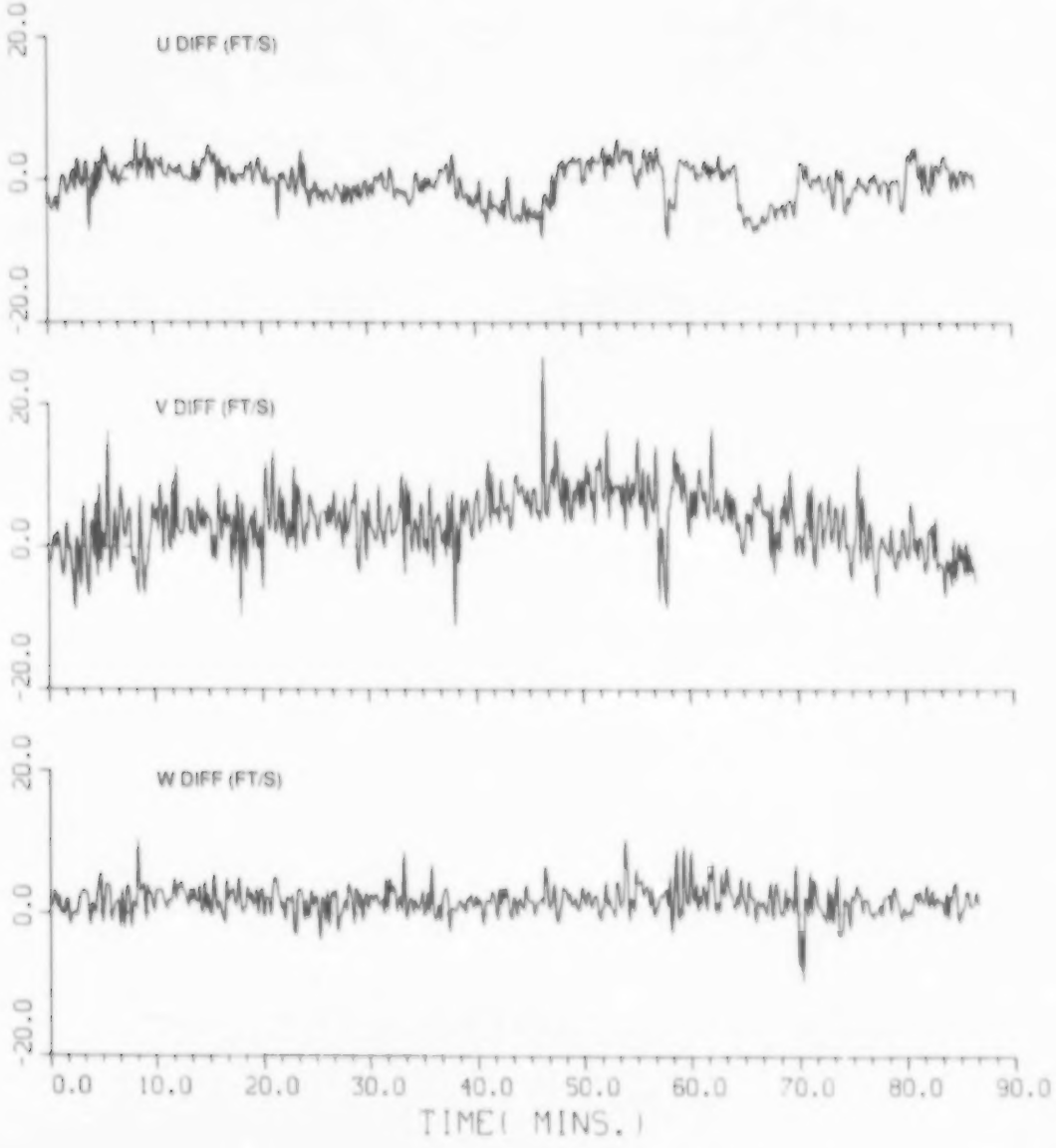

FIG. 29: DOPPLER VELOCITY MEASUREMENTS FOR FLIGHT #1

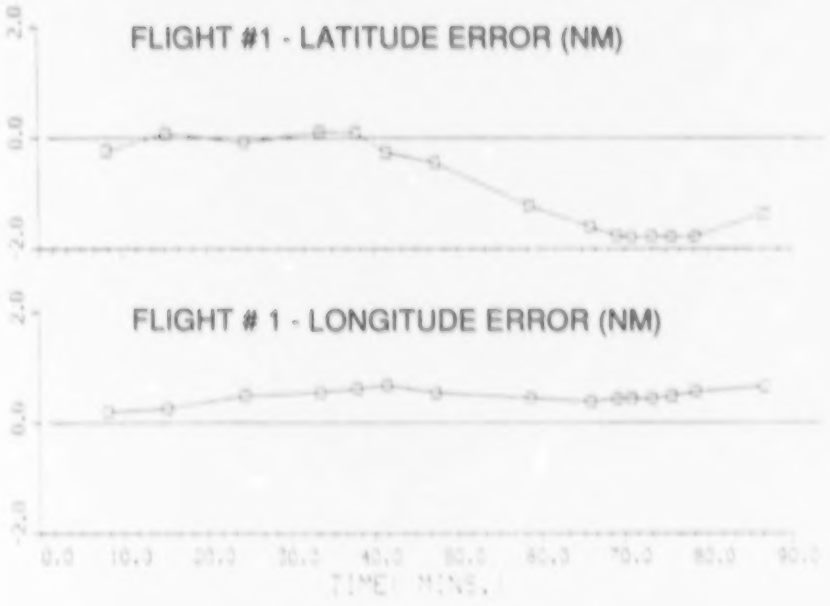

FIG. 30: IRS POSITION ERRORS AT VOT'S FOR FLIGHT #1

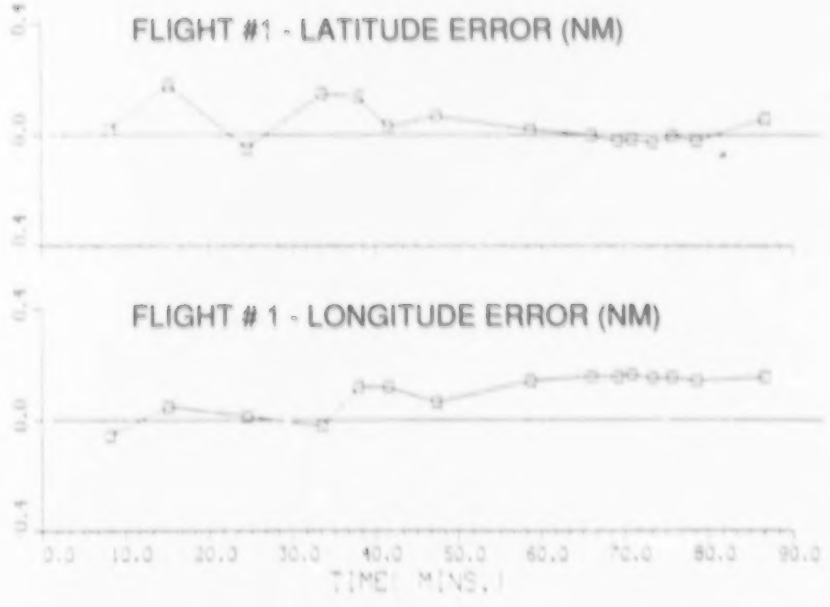

FIG. 31: LORAN-C POSITION ERRORS AT VOT'S FOR FLIGHT #1

```
0000                  - NO. OF SECONDS OF INITIAL DATA TO BE SKIPPED
05210                 - DURATION OF KF RUN (IN SECONDS, DIVISIBLE BY 10)
00000                 - START TIME FOR OVERWATER FLIGHT (IN SECS RELATIVE)
00000                 - END TIME FOR OVERWATER FLIGHT   (TO START OF FILE)
45.000                - NOMINAL OPERATING LATITUDE (DEGS)
-06.63 000.92 001.48  - U,V,W ACCELER-TO-INS LEVER ARMS (FEET)
* INS SYSTEM ERROR STATE STATISTICS (1-SIGMA):
1.   0.2000E00 2.   0.2000E00  LAT & LONG (NM)
3.   3.0000E01 4.   0.3300E01  ALT (FT) & N. VELOC. (FT/SEC)
5.   0.3300E01 6.   0.3300E01  EAST & VERT. VELOC'S (FT/SEC)
7.   0.3340E-1 8.   0.3340E-1  N. & E. AXIS ATTITUDE (DEG)
9.   0.8594E-1 10.  0.2500E-1  Z AXIS ATT. (DEG) & BARO LOOP ACC. (FT/S**2)
* SENSOR MARKOV ERROR STATISTICS (1-SIGMA VALUE & CORR. TIME IN SECS.):
11.  2.0000E-2   07500.0       X AXIS GYRO (DEG/HR)
12.  2.0000E-2   07500.0       Y AXIS GYRO (DEG/HR)
13.  2.0000E-2   07500.0       Z AXIS GYRO (DEG/HR)
14.  1.0000E-2   07500.0       N AXIS ACCEL (FT/SEC**2)
15.  2.0000E-3   15000.0       Y AXIS ACCEL (FT/SEC**2)
16.  2.0000E-3   15000.0       Z AXIS ACCEL (FT/SEC**2)
17.  2.0000E02   05000.0       ALTIMETER BIAS (FEET)
18.  0.2000E-0   03600.0       LATITUDE LORAN-C FIX (NM)
19.  0.2000E-0   03600.0       LONGITUDE LORAN-C FIX (NM)
20.  0.0200E00   03600.0       DOPPLER SCALE FACTOR
21.  2.3000E00   03600.0       DOPPLER V BORESIGHT (DEG)
22.  1.1500E00   03600.0       DOPPLER H BORESIGHT (DEG)
23.  4.5000E00   00900.0       NORTH SEA BIAS (FT/SEC)
24.  4.5000E00   00900.0       EAST SEA BIAS (FT/SEC)
* INS SYSTEM PLANT NOISE STATISTICS (1-SIGMA):
1.   2.3000E-6 2.   2.3000E-6  INTEGRATED X & Y GYRO (DEG)
3.   2.3000E-6 4.   1.0000E-3  INTEGRATED Z GYRO (DEG) & X ACC. (FT/SEC)
5.   1.0000E-3 6.   1.0000E-3  INTEGRATED Y & Z ACCEL. (FT/SEC)
7.   1.0000E-3 8.   1.0000E-3  INTEGRATED N & E GRAVITY (FT/SEC)
9.   1.0000E-3 10.  3.1620E01  INTEG'D Z GRAVITY (FT/S) & ALT NOISE (FT-S)
* MEASUREMENT NOISE STATISTICS (UNAVERAGED, 1-SIGMA):
1.   0.0700E00 2.   0.0700E00  LAT & LONG LORAN-C (NM)
3.   5.3400E00 4.   1.0680E01 5.   5.3400E00  U,V,W DOPPLER (FT/SEC)
6.   5.0000E00                  BAROMETRIC ALTIMETER (FEET)
* VISUAL ON-TOP REFERENCE INFORMATION:
0 15                             - POS'N FIX FLAG & TOTAL NO. OF POINTS
1.000E-1 1.000E-1                - 1 SIGMA LEVEL (NM) FOR LAT & LONG FIX
0490    45.44135    75.89501   1 - TIME, LAT/LONG & FLAG FOR EACH ON-TOP
0910    45.47167    76.23333   1
1470    45.58434    76.67900   1
2010    45.51082    76.89049   1
2270    45.19934    76.90734   1
2490    45.24216    76.76033   1
2840    45.38618    76.57565   1
3520    45.18333    76.12667   1
3960    45.07417    75.90467   1
4150    45.16051    75.83766   1
4250    45.16008    75.83814   1
4400    45.08076    75.90078   1
4540    45.07919    75.90170   1
4710    45.15884    75.83893   1
5200    45.31551    75.66766   1
```

FIG. 32: SDKFNAV CTRL FILE FOR FLIGHT #1

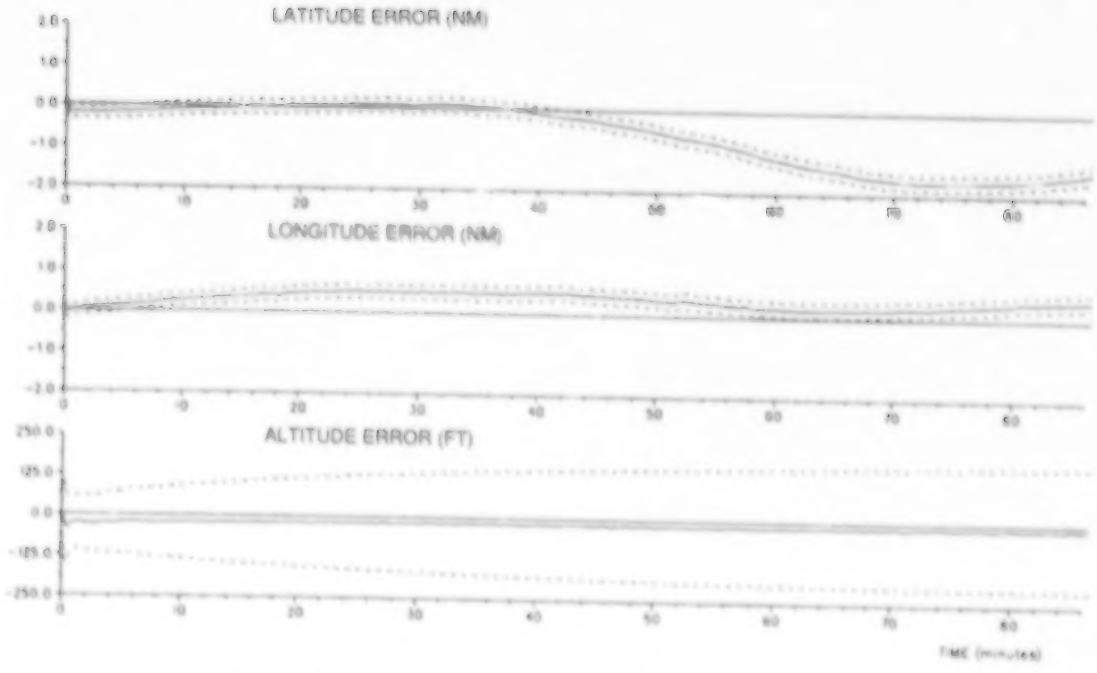

FIG. 33: IRS POSITION ERROR STATE ESTIMATES FOR FLIGHT #1

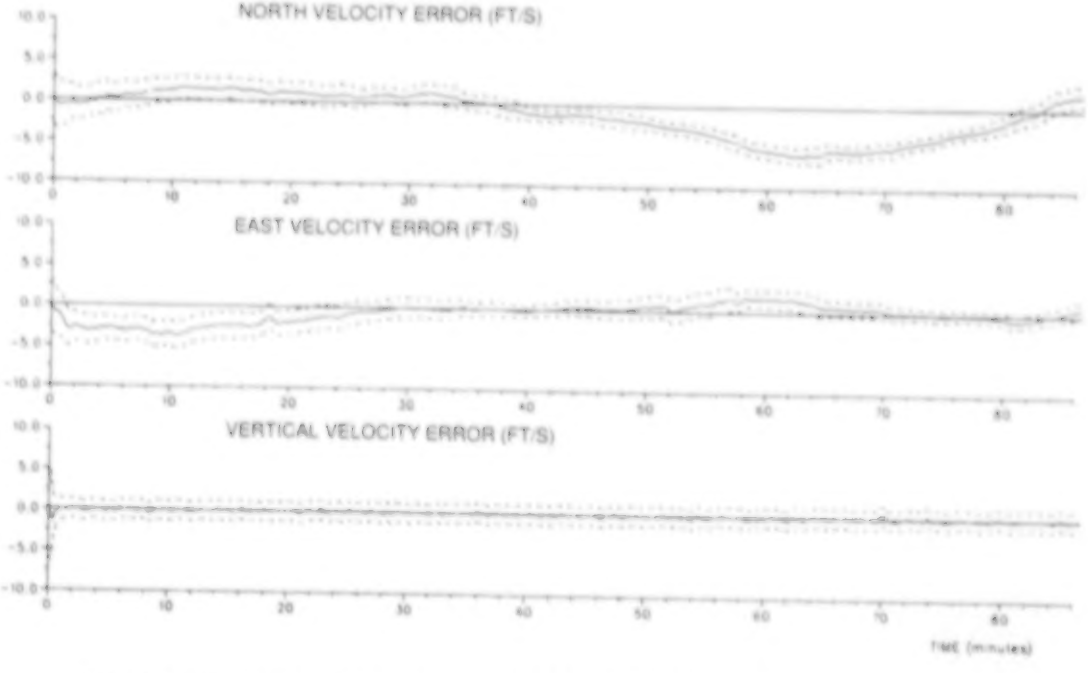

FIG. 34: IRS VELOCITY ERROR STATE ESTIMATES FOR FLIGHT #1

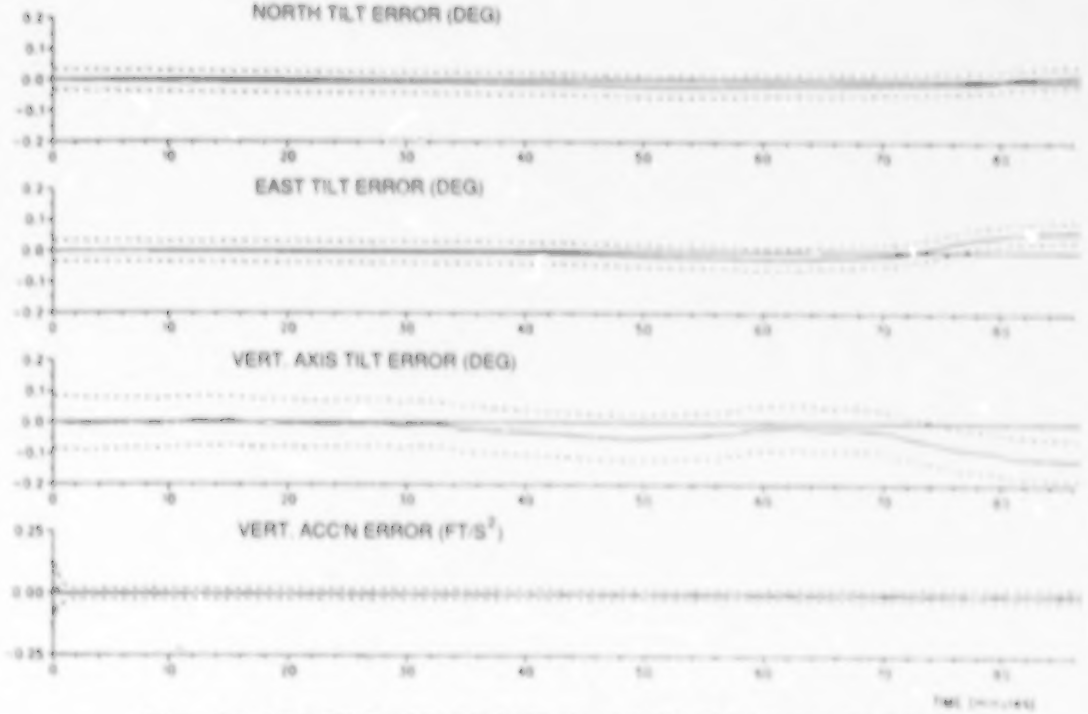

FIG. 35: IRS TILT AND VERTICAL ACCELERATION ERROR STATE ESTIMATES FOR FLIGHT #1

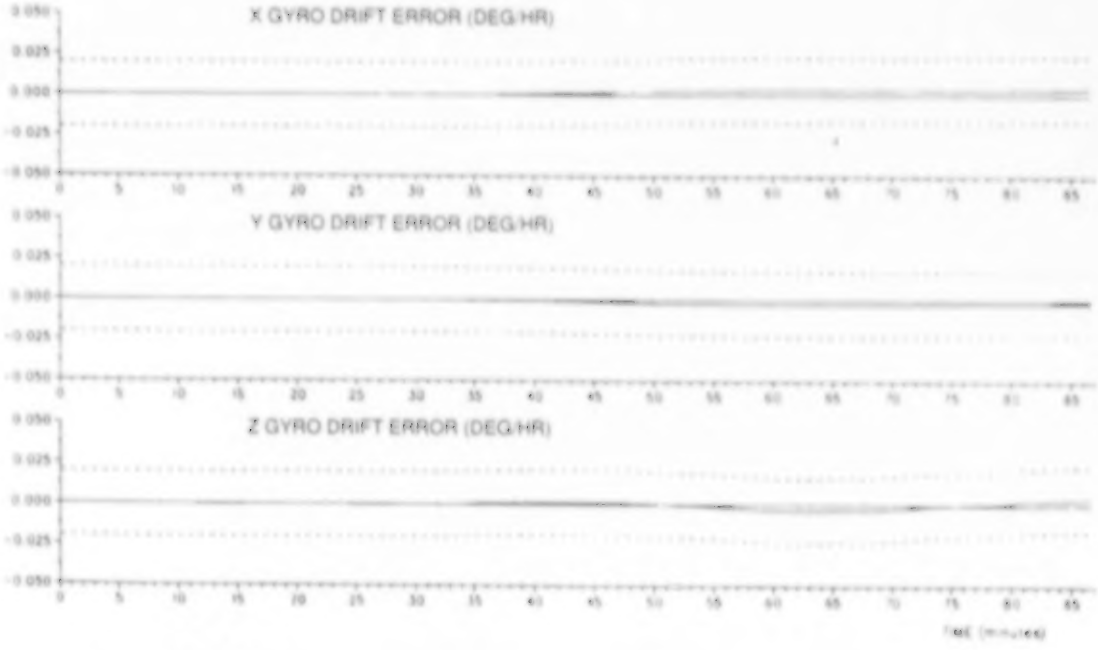

FIG. 36: IRS GYRO DRIFT ERROR STATE ESTIMATES FOR FLIGHT #1

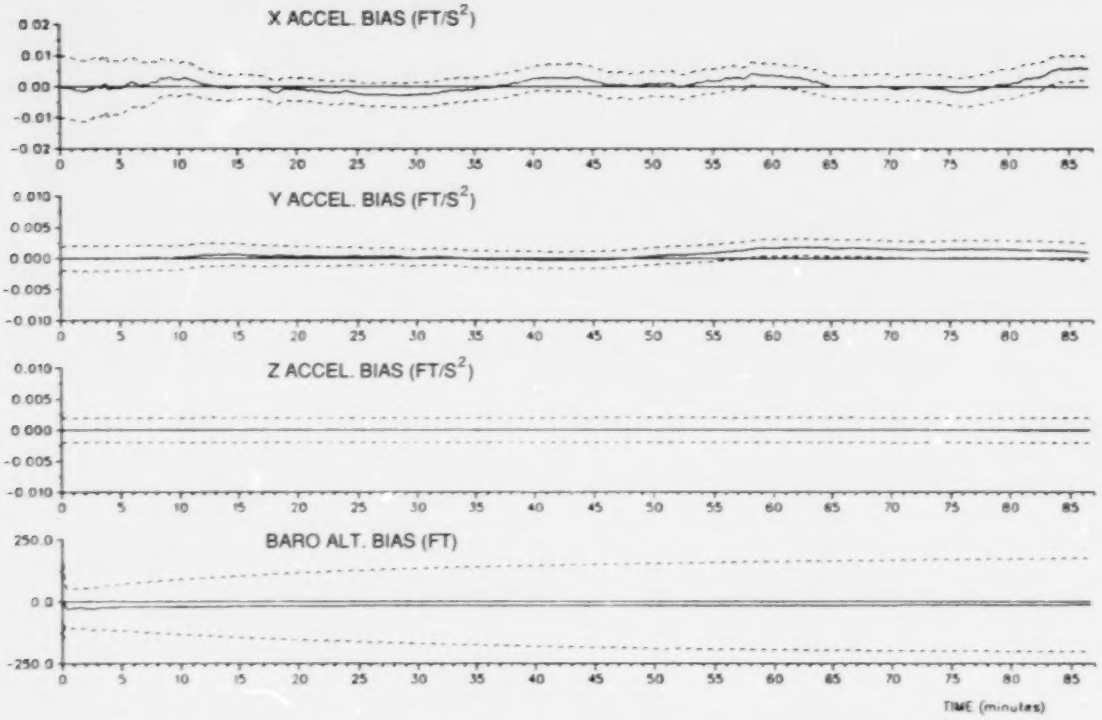

FIG. 37: IRS ACCELEROMETER AND BAROMETRIC ALTIMETER ERROR STATE ESTIMATES FOR FLIGHT #1

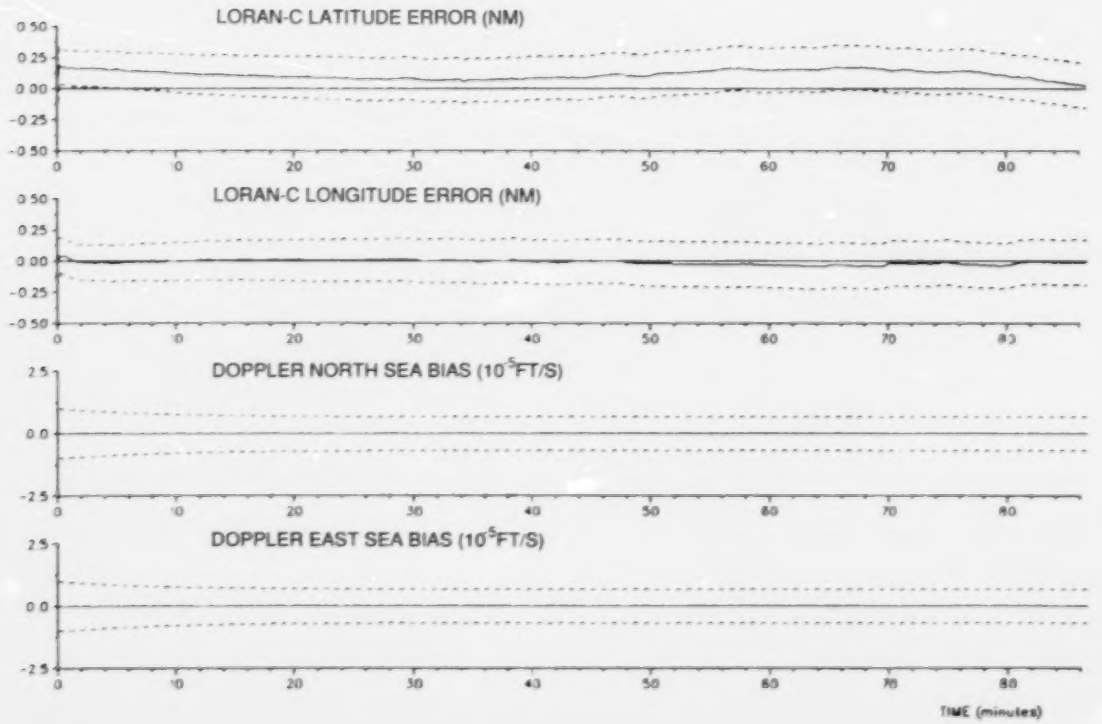
LORAN-C LATITUDE ERROR (NM)
LORAN-C LONGITUDE ERROR (NM)
DOPPLER NORTH SEA BIAS (10^{-5}FT/S)
DOPPLER EAST SEA BIAS (10^{-5}FT/S)
TIME (minutes)

FIG. 38: LORAN-C POSITION AND DOPPLER SEA BIAS ERROR STATE ESTIMATES FOR FLIGHT #1

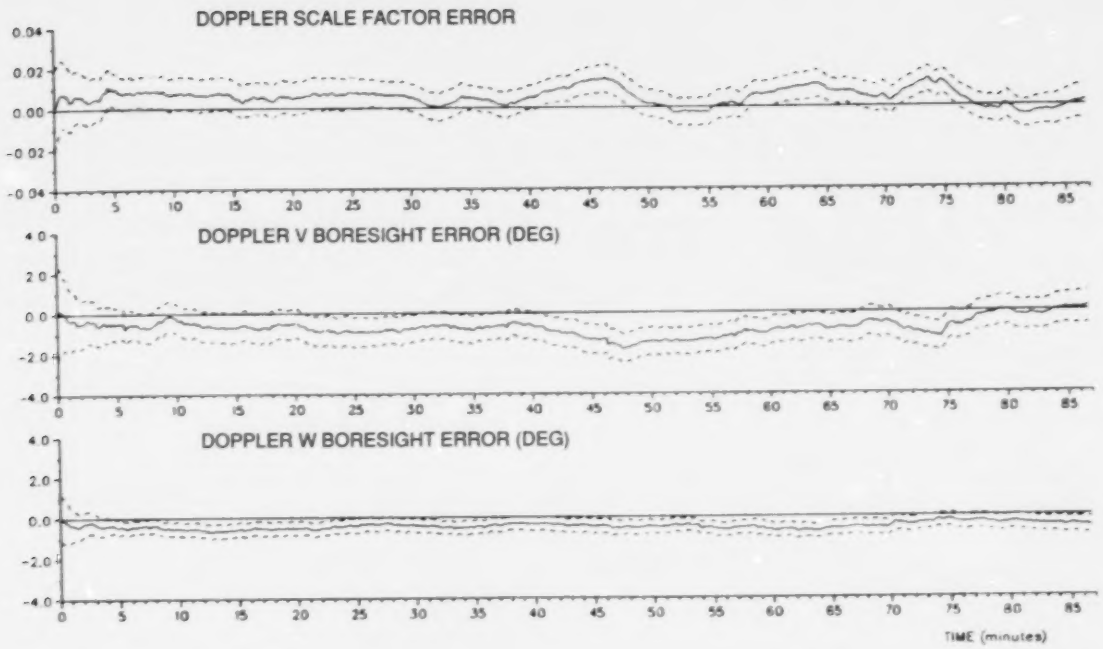

FIG. 39: DOPPLER VELOCITY ERROR STATE ESTIMATES FOR FLIGHT #1

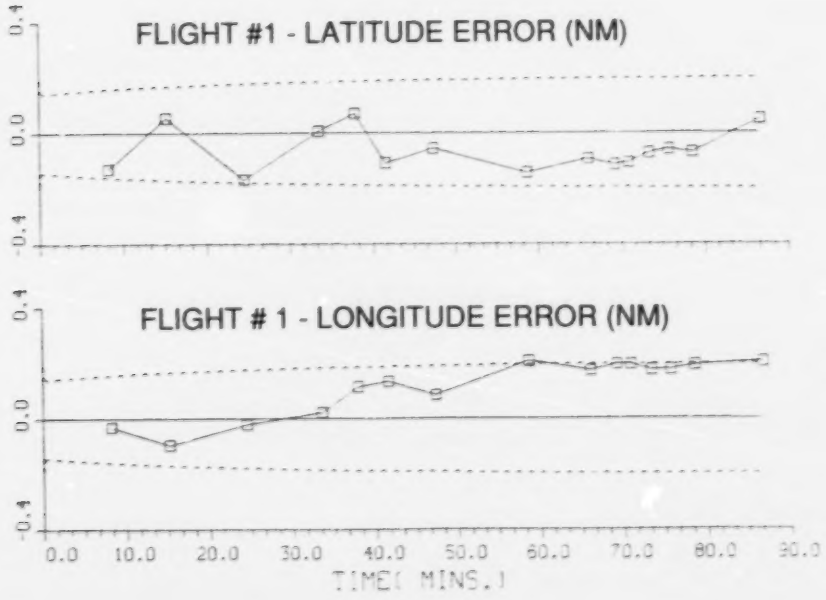

FIG. 40: KALMAN-CORRECTED IRS POSITION ERRORS AT VOT'S FOR FLIGHT #1

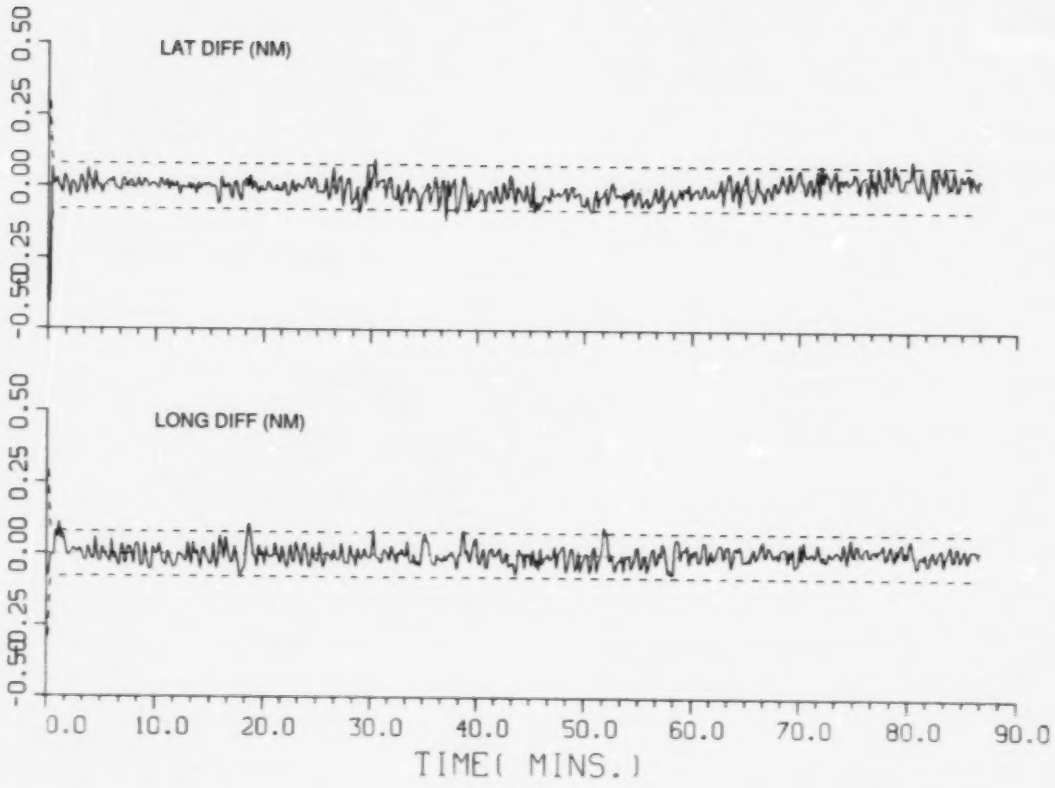

FIG. 40a: INNOVATIONS FOR POSITION MEASUREMENTS FROM FLIGHT #1

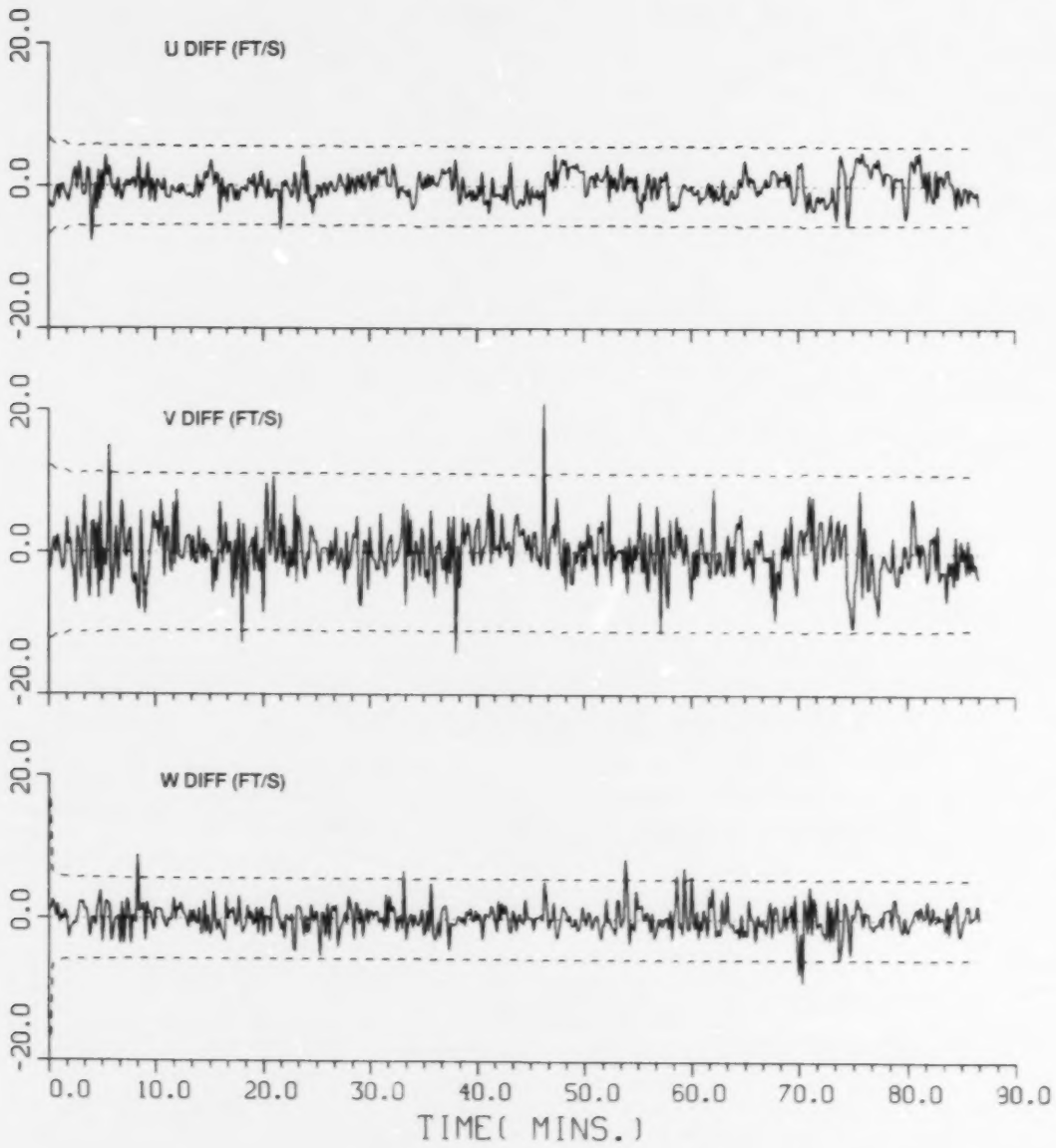

FIG. 40b: INNOVATIONS FOR VELOCITY MEASUREMENTS FROM FLIGHT #1

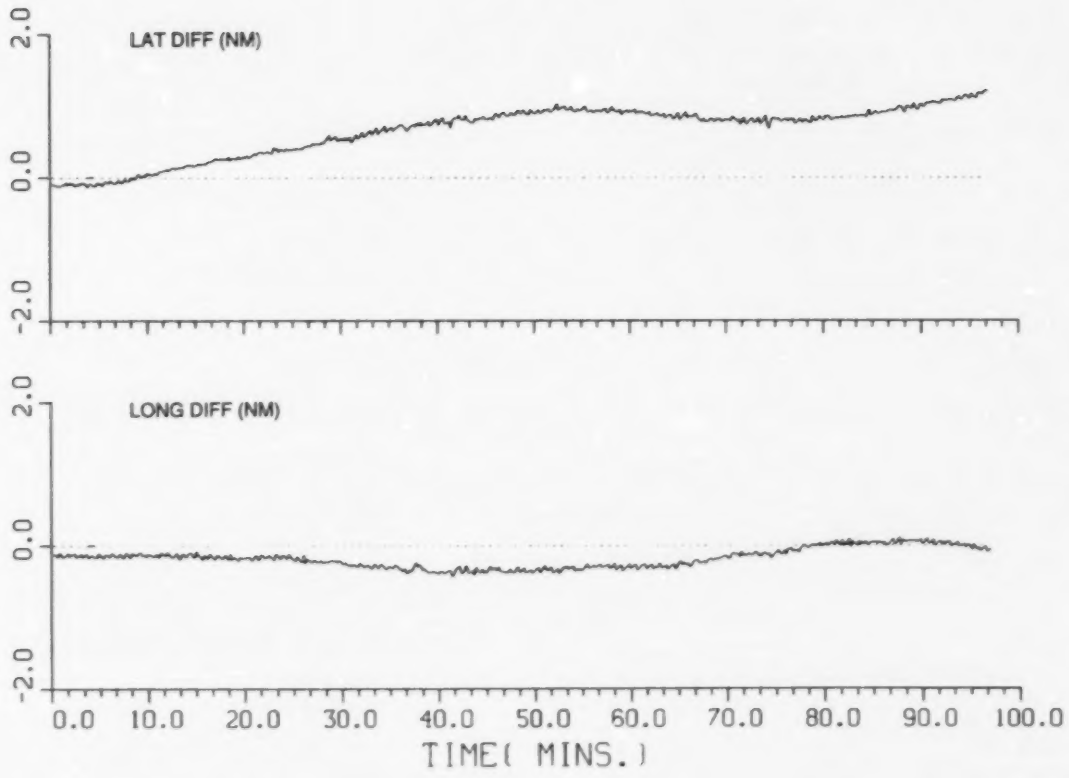

FIG. 41: LORAN-C POSITION MEASUREMENTS FOR FLIGHT #2

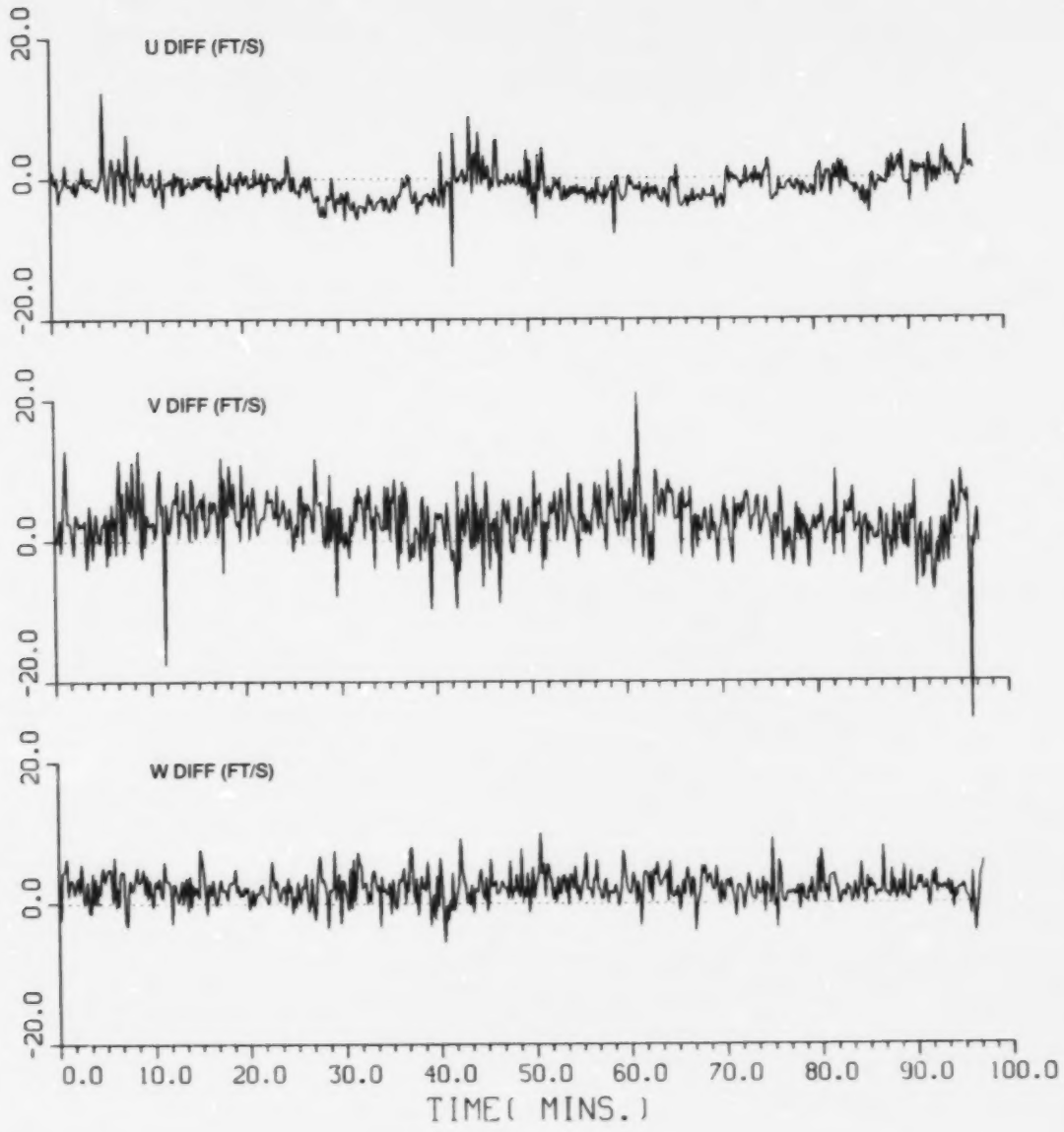

FIG. 42: DOPPLER VELOCITY MEASUREMENTS FOR FLIGHT #2

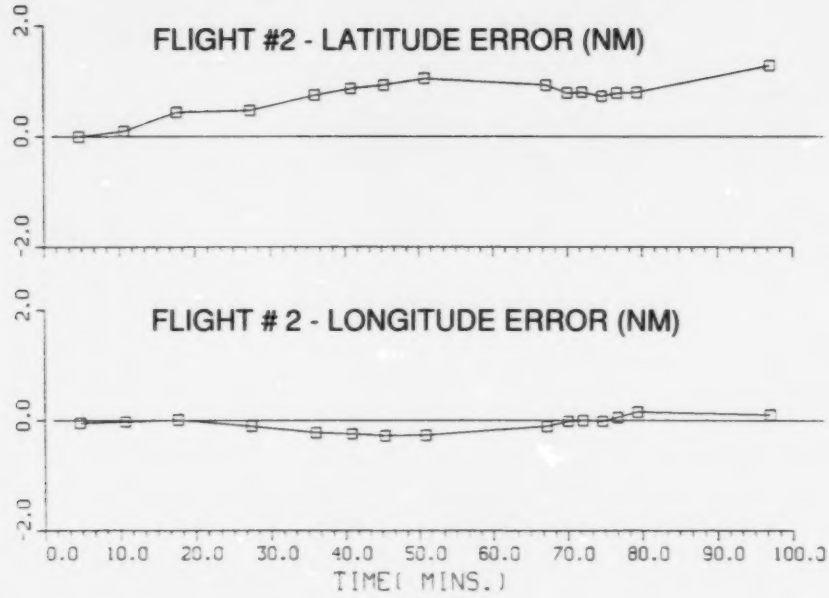

FIG. 43: IRS POSITION ERRORS AT VOT'S FOR FLIGHT #2

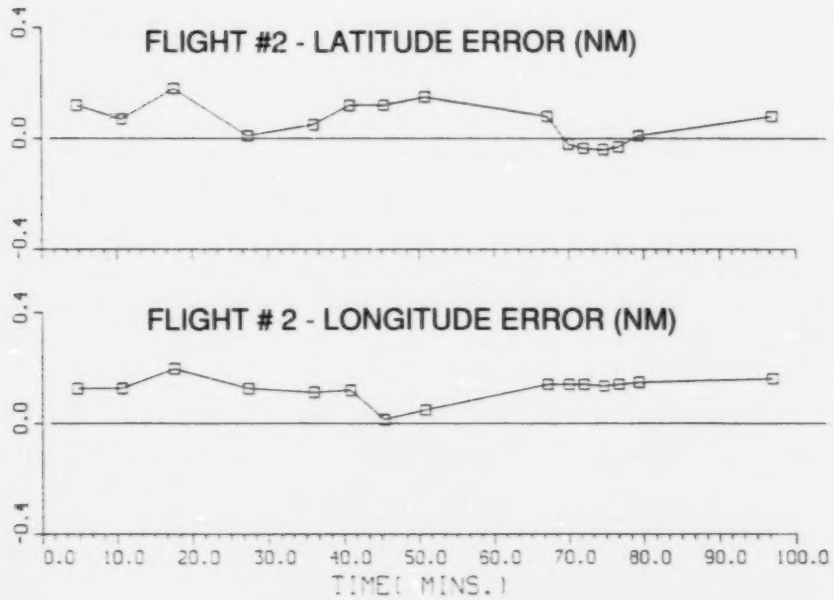

FIG. 44: LORAN-C POSITION ERRORS AT VOT'S FOR FLIGHT #2

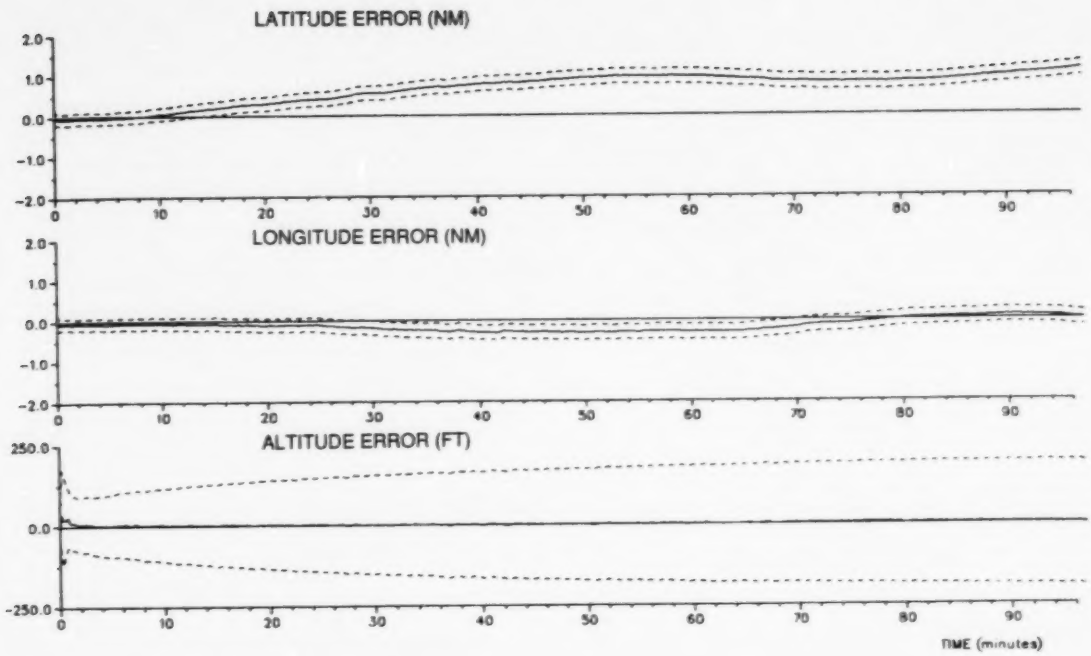

FIG. 45: IRS POSITION ERROR STATE ESTIMATES FOR FLIGHT #2

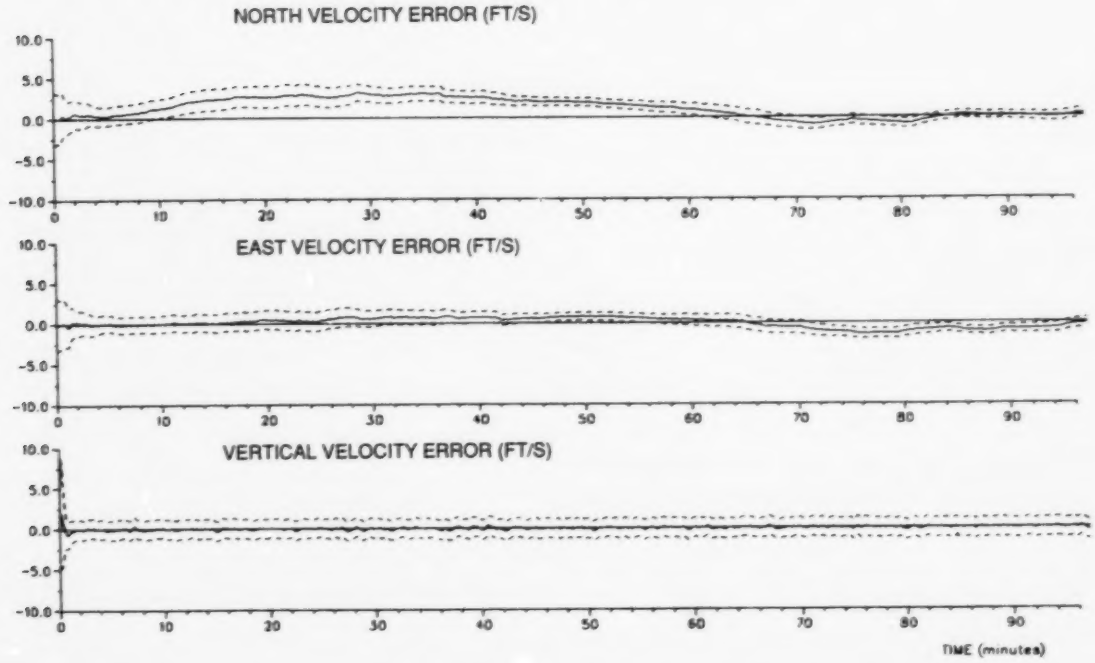

FIG. 46: IRS VELOCITY ERROR STATE ESTIMATES FOR FLIGHT #2

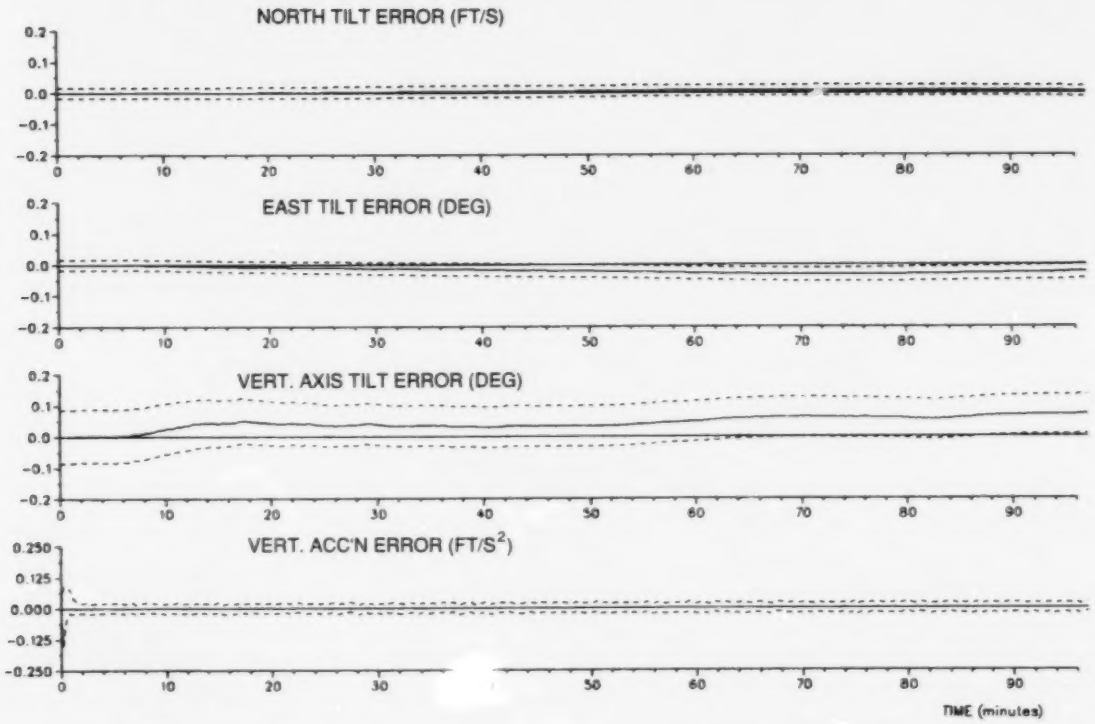

FIG. 47: IRS TILT AND VERTICAL ACCELERATION ERROR STATE ESTIMATES FOR FLIGHT #2

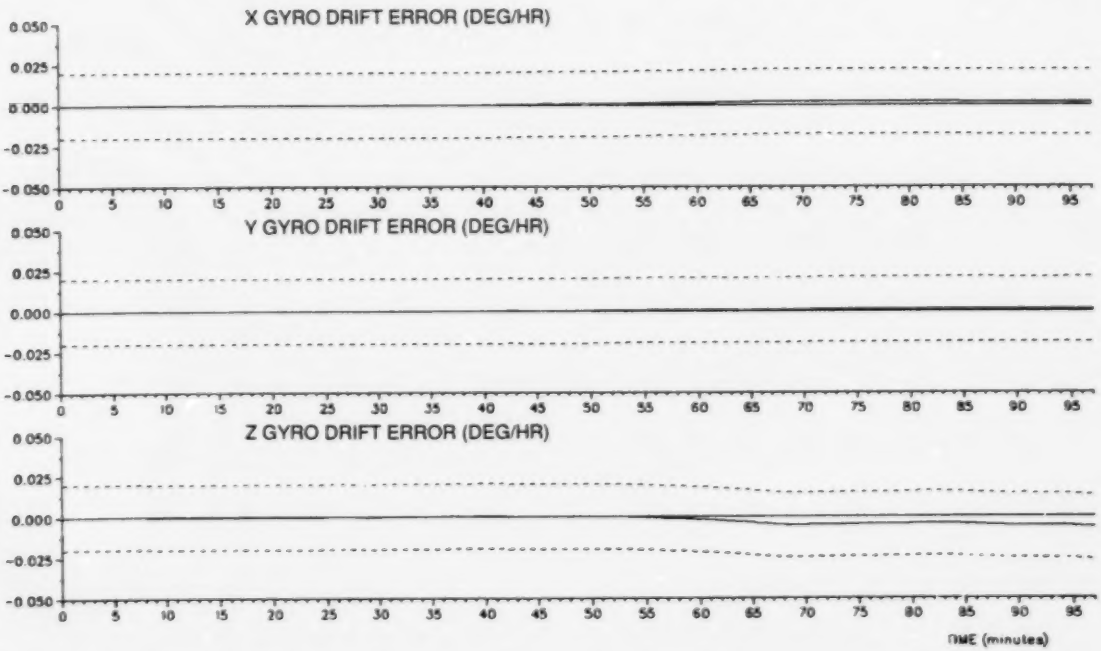

FIG. 48: IRS GYRO DRIFT ERROR STATE ESTIMATES FOR FLIGHT #2

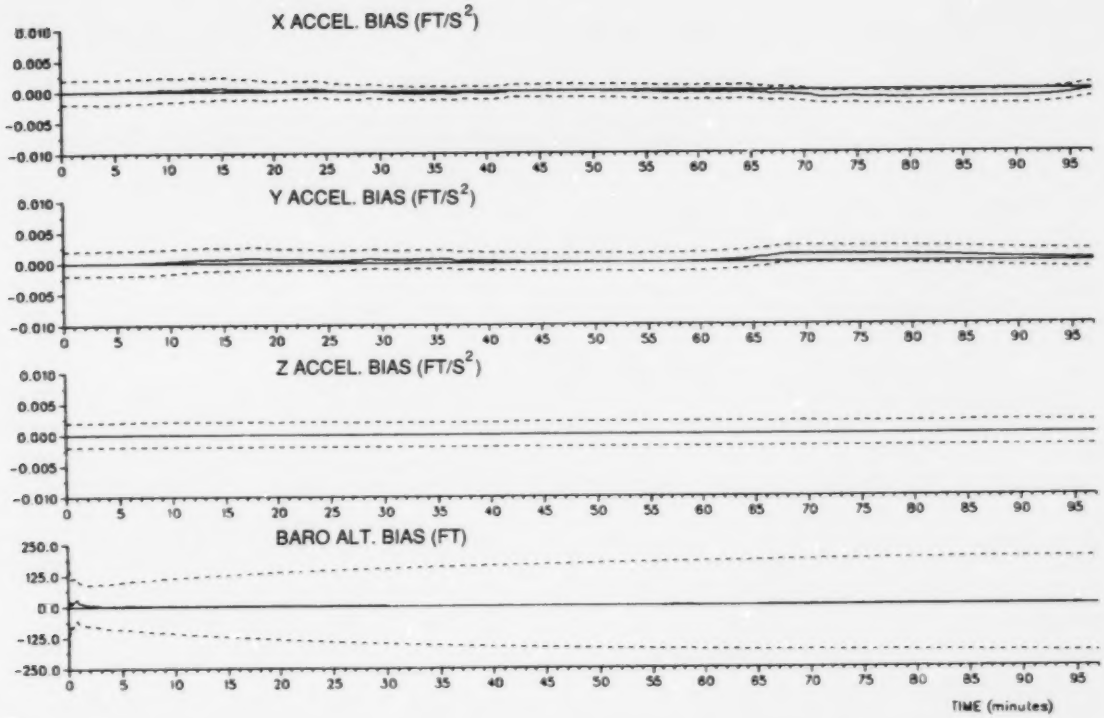

FIG. 49: IRS ACCELEROMETER AND BARO ALTIMETER ERROR STATE ESTIMATES FOR FLIGHT #2

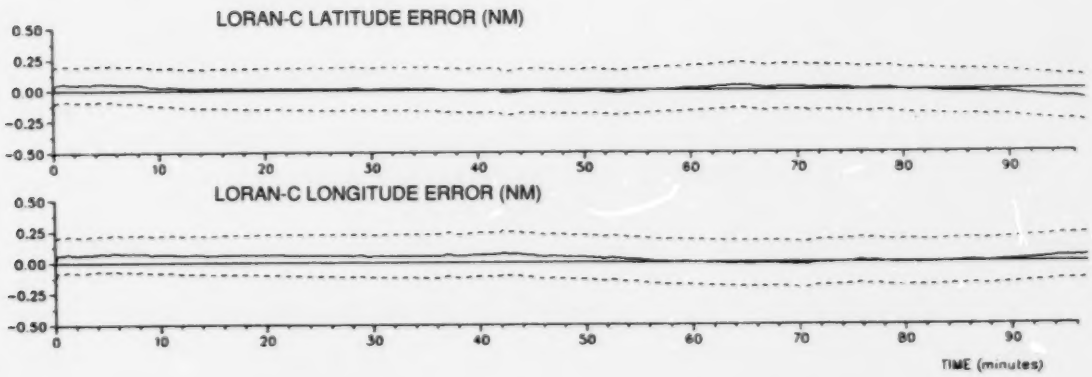

FIG. 50: LORAN-C POSITION ERROR STATE ESTIMATES FOR FLIGHT # 2

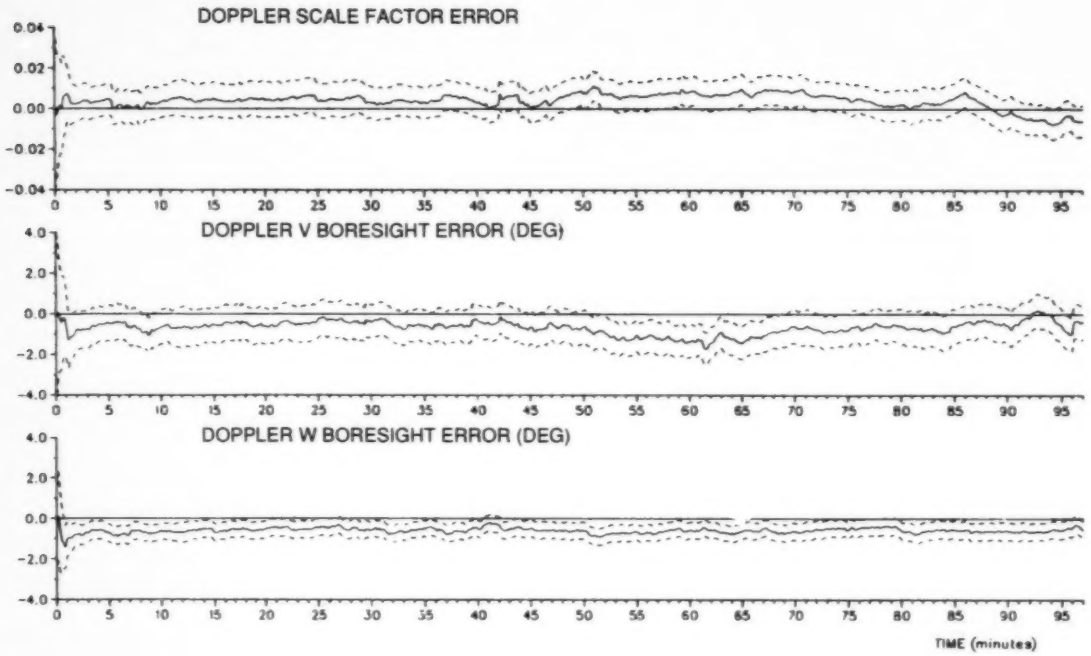

FIG. 51: DOPPLER VELOCITY ERROR STATE ESTIMATES FOR FLIGHT #2

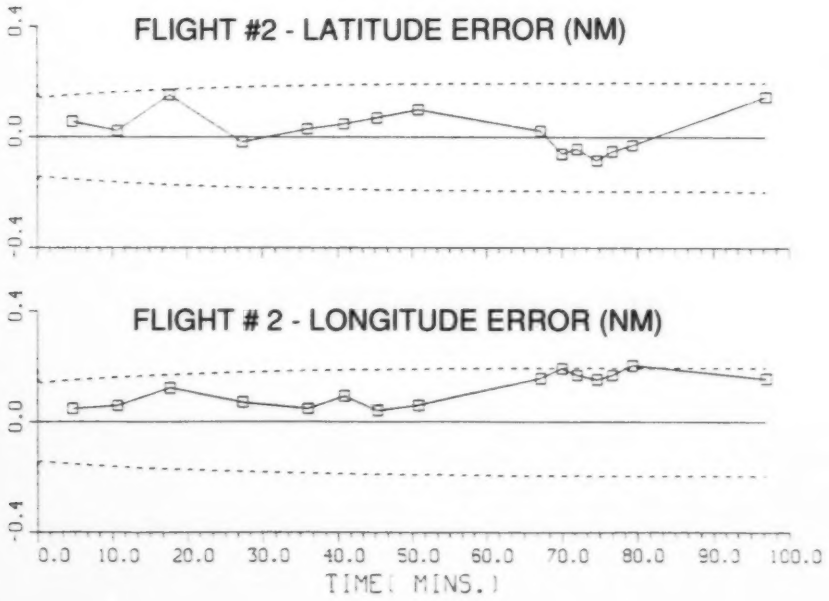

FIG. 52: KALMAN-CORRECTED IRS POSITION ERRORS AT VOT'S FOR FLIGHT #2

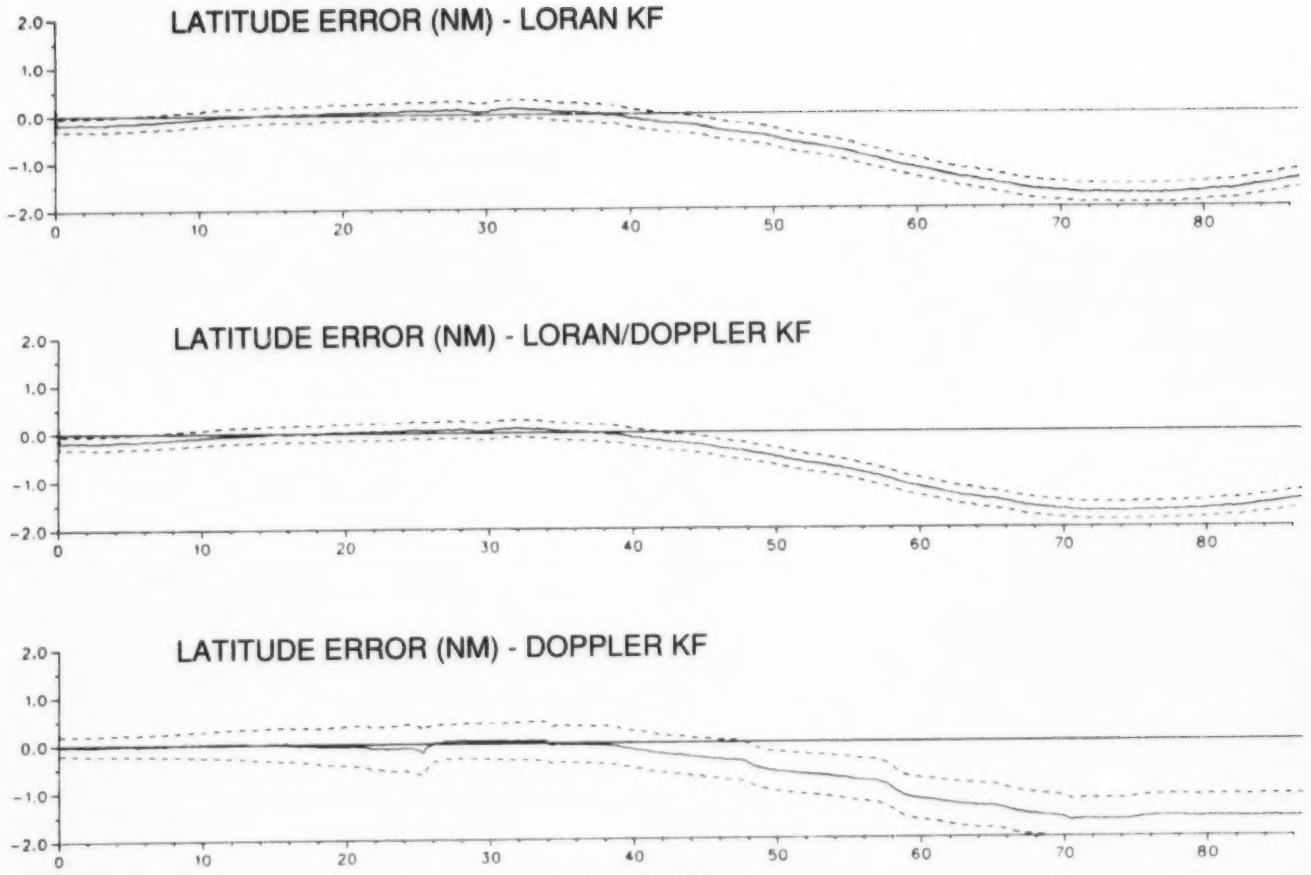

FIG. 53: COMPARISON OF KALMAN FILTER CONFIGURATIONS - LATITUDE ERRORS FOR FLIGHT #1

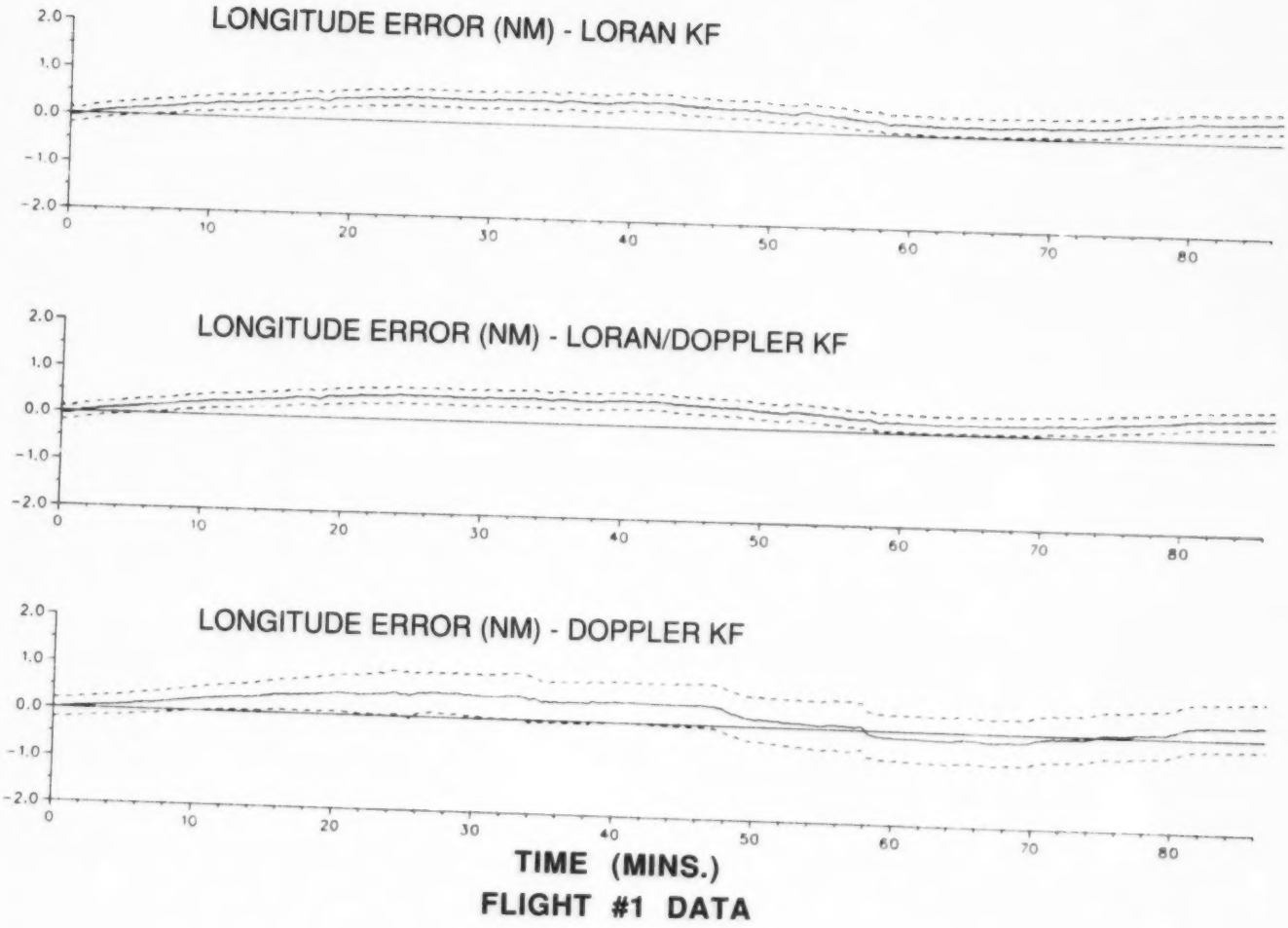

FIG. 54: COMPARISON OF KALMAN FILTER CONFIGURATIONS - LONGITUDE ERRORS FOR FLIGHT #1

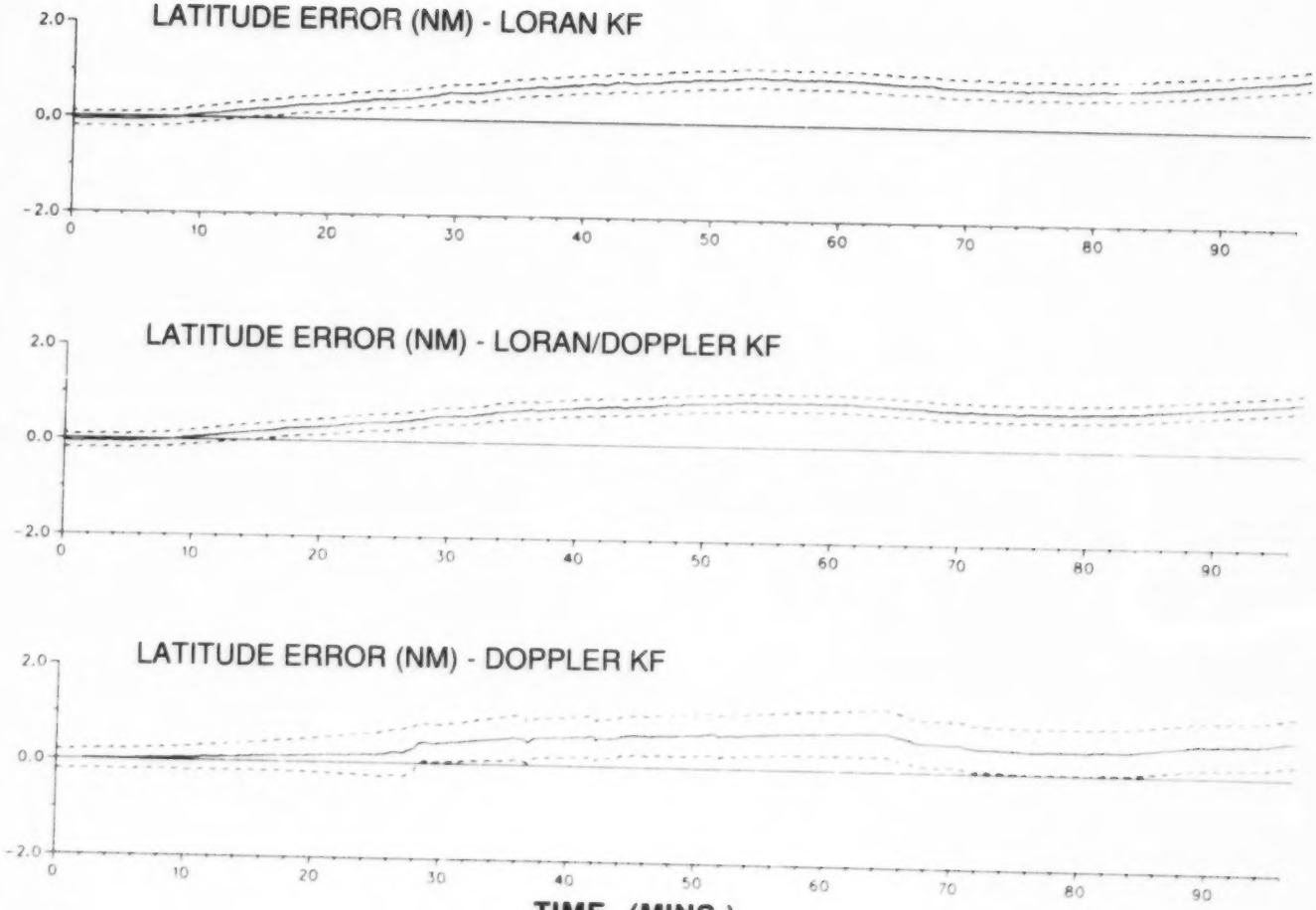

FIG. 55: COMPARISON OF KALMAN FILTER CONFIGURATIONS - LATITUDE ERRORS FOR FLIGHT #2

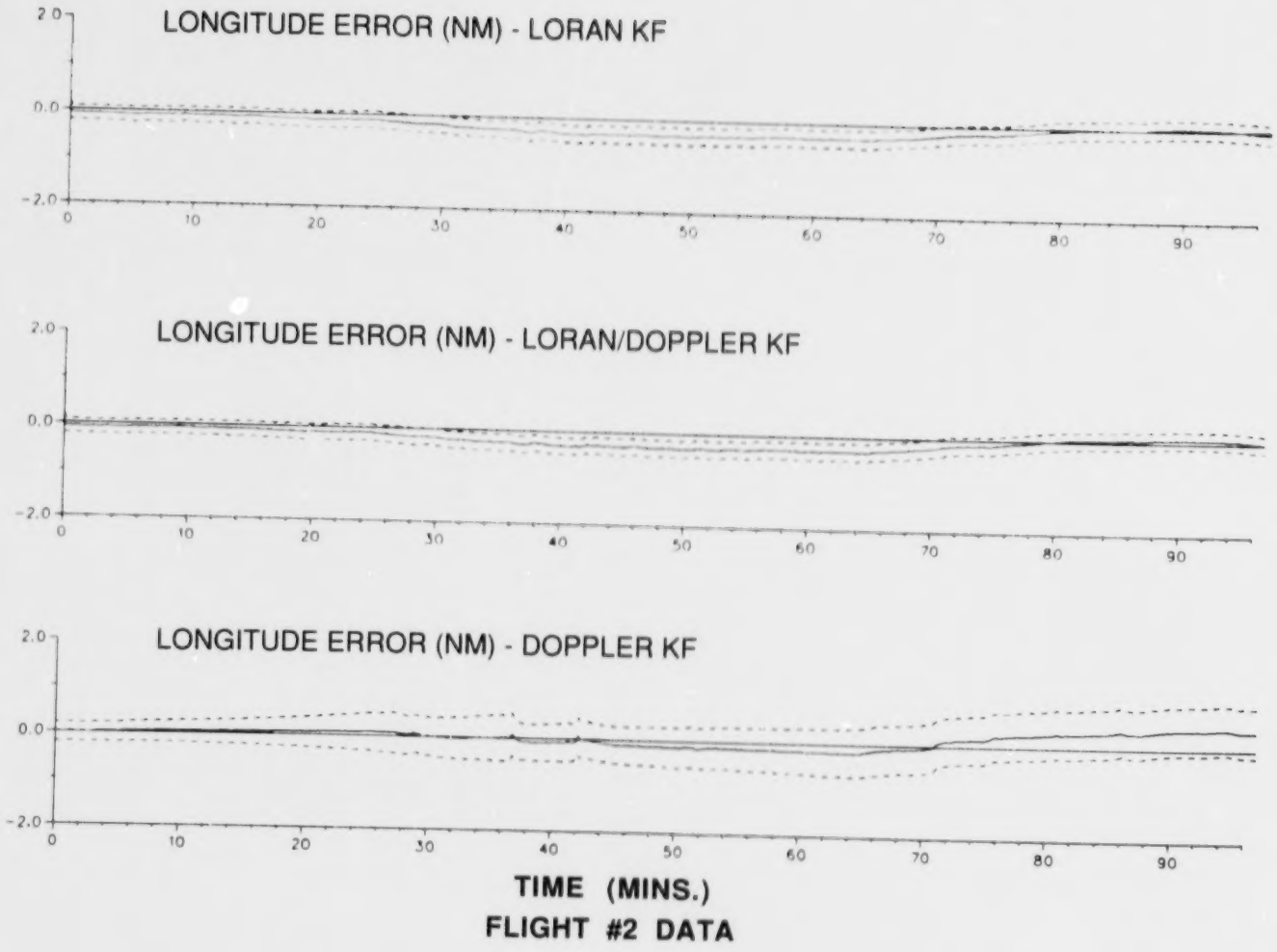

FIG. 56: COMPARISON OF KALMAN FILTER CONFIGURATIONS - LONGITUDE ERRORS FOR FLIGHT #2

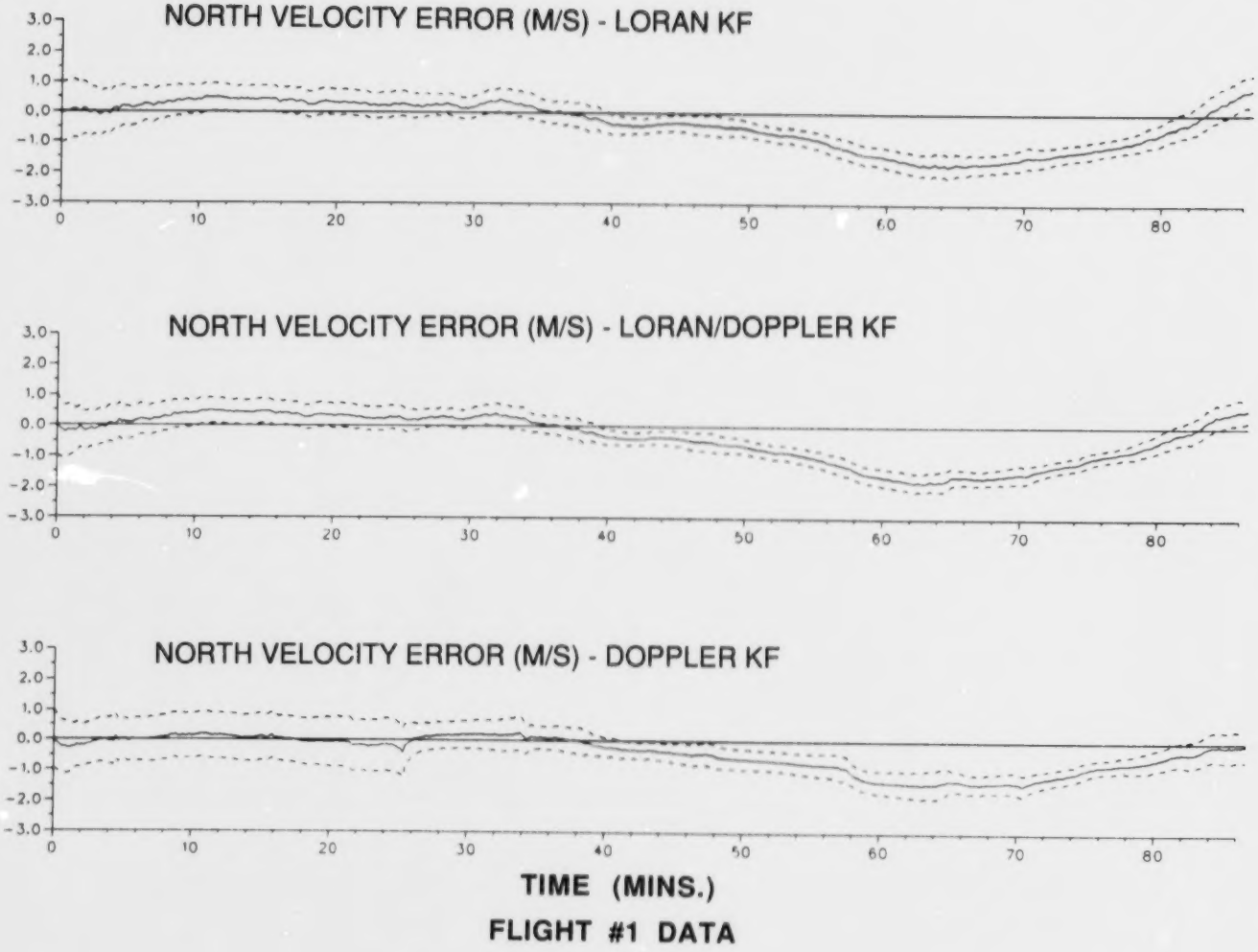

FIG. 57: COMPARISON OF KALMAN FILTER CONFIGURATIONS - NORTH VELOCITY ERRORS FOR FLIGHT #1

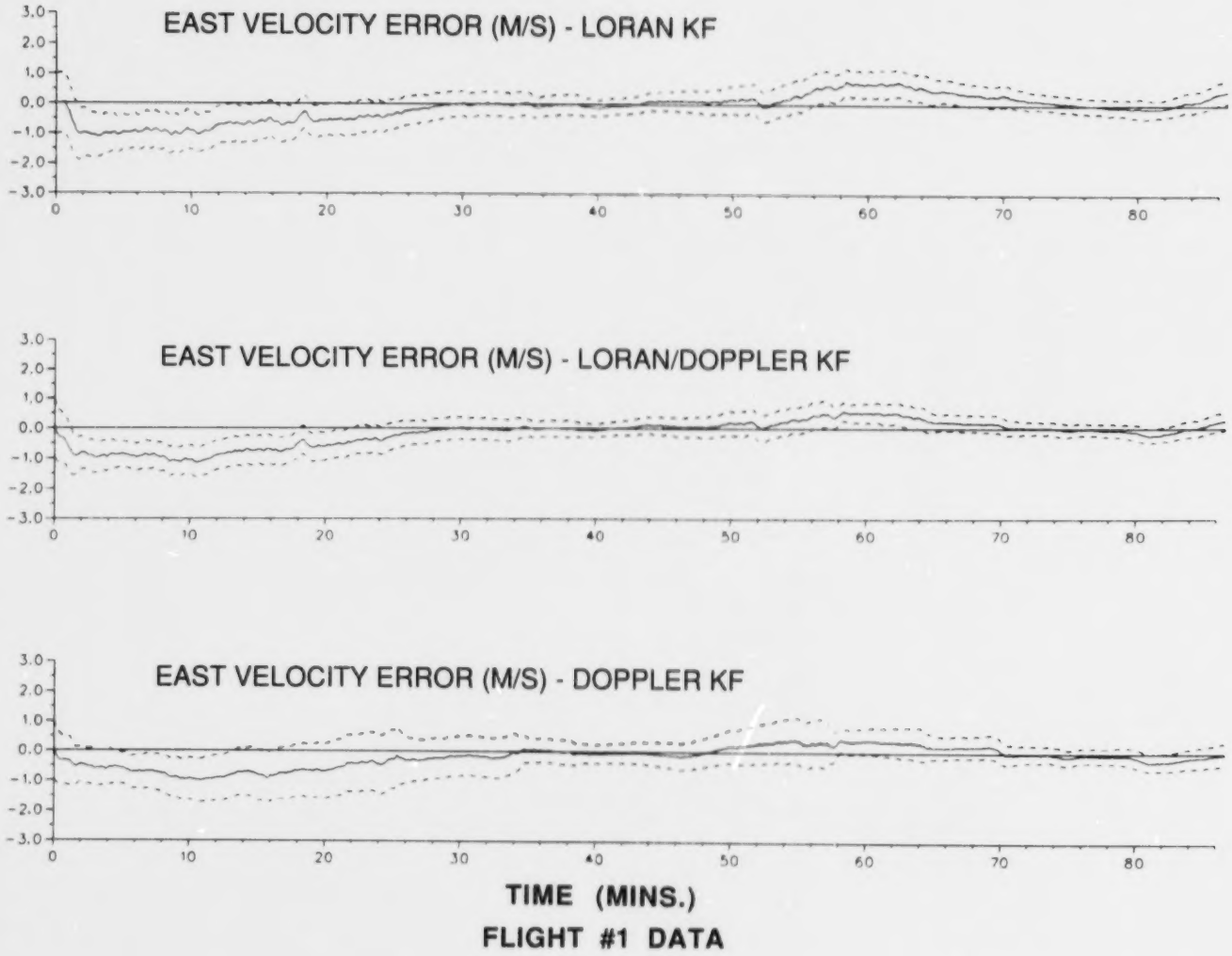

FIG. 58: COMPARISON OF KALMAN FILTER CONFIGURATIONS - EAST VELOCITY ERRORS FOR FLIGHT #1

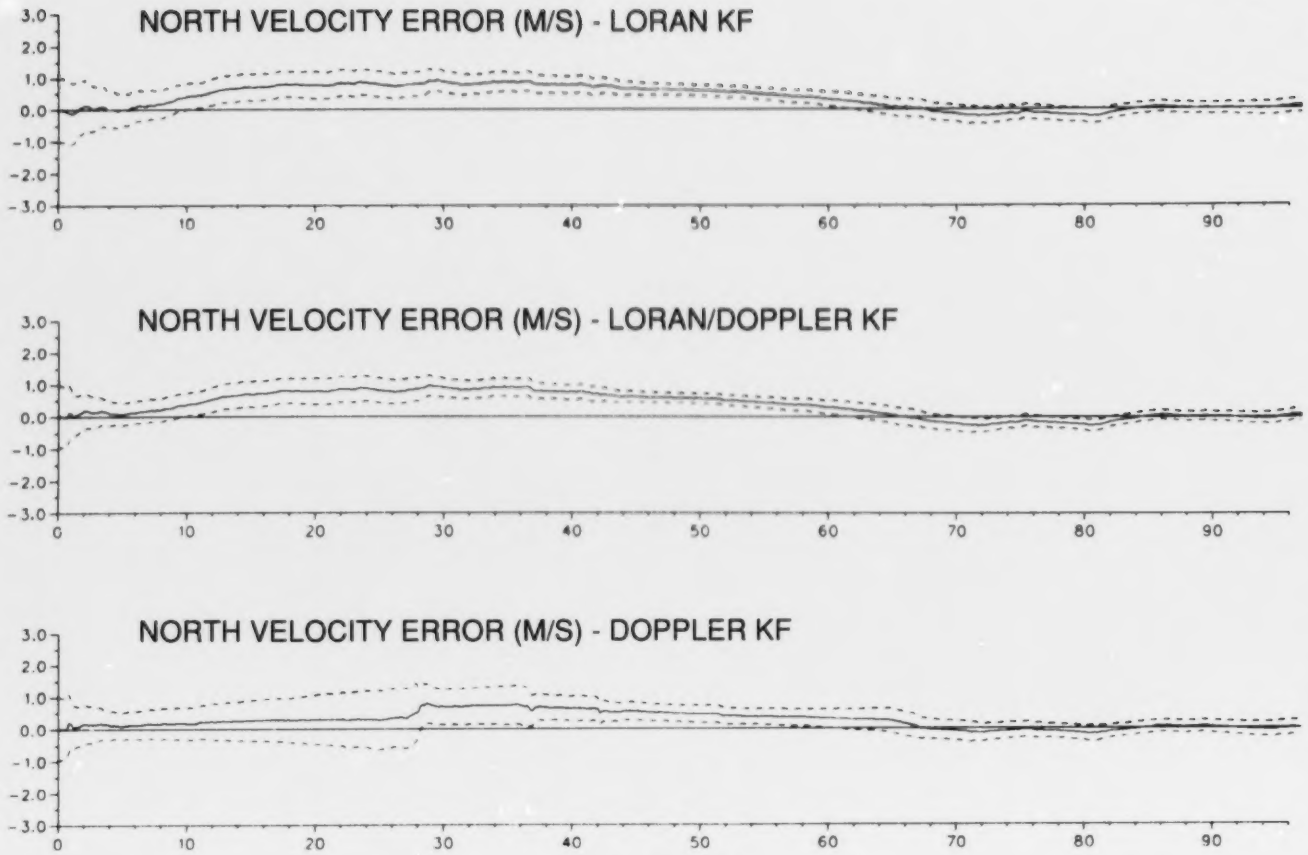

FIG. 59: COMPARISON OF KALMAN FILTER CONFIGURATIONS - NORTH VELOCITY ERRORS FOR FLIGHT #2

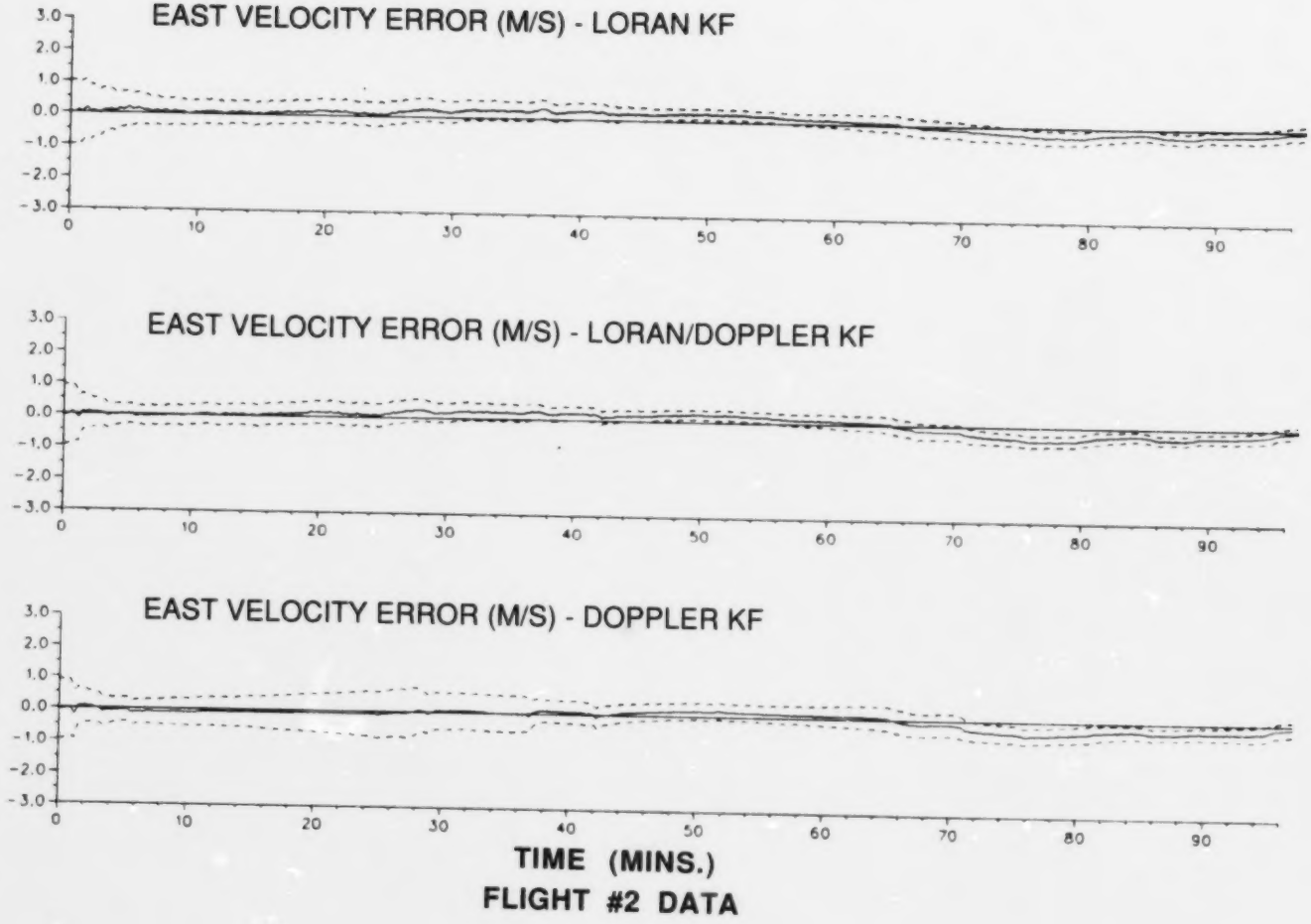

FIG. 60: COMPARISON OF KALMAN FILTER CONFIGURATIONS - EAST VELOCITY ERRORS FOR FLIGHT #2

REPORT DOCUMENTATION PAGE / PAGE DE DOCUMENTATION DE RAPPORT

REPORT/RAPPORT IAR-AN-72 1a	REPORT/RAPPORT NRC No. 32148 1b
REPORT SECURITY CLASSIFICATION CLASSIFICATION DE SÉCURITÉ DE RAPPORT Unclassified 2	DISTRIBUTION (LIMITATIONS) Unlimited 3

TITLE/SUBTITLE/TITRE/SOUS-TITRE

A Kalman Filter Integrated Navigation Design
for the IAR Twin Otter Atmospheric Research Aircraft

4

AUTHOR(S)/AUTEUR(S)

B.W. Leach

5

SERIES/SÉRIE

Aeronautical Note

6

CORPORATE AUTHOR/PERFORMING AGENCY/AUTEUR D'ENTREPRISE/AGENCE D'EXÉCUTION

National Research Council Canada
Institute for Aerospace Research — Flight Research Laboratory

7

SPONSORING AGENCY/AGENCE DE SUBVENTION

8

DATE	FILE/DOSSIER	LAB. ORDER COMMANDE DE LAB.	PAGES	FIGS./DIAGRAMMES
04-91 9	10	11	100 12a	60 12b

NOTES

13

DESCRIPTORS (KEY WORDS)/MOTS-CLÉS

14
1. Kalman Filters - Navigation Aids
2. Navigation Aids - Kalman Filters
3. Twin Otter (IAR) - Atmospheric Research Aircraft

SUMMARY/SOMMAIRE

The IAR Twin Otter Atmospheric Research Aircraft has a continuing requirement for more accurate, inertially-based navigation data for both track recovery and the calculation of wind gust components. This navigational accuracy is necessary, not just during post-flight analysis, but also for real-time, in-flight guidance and wind computation. Previous developmental work at the Flight Research Laboratory on advanced navigation systems has demonstrated the benefits of a Kalman filter integrated navigation approach in order to satisfy the most stringent navigational requirements. A significant upgrade to the navigation sensor suite onboard the Twin Otter in the last two years has resulted in the potential, via Kalman filtering, for generating very high quality inertial velocity and positional information in real time, together with improved airborne wind components.

15